NASA-CR-195918

AN ARBITRARY GRID CFD ALGORITHM FOR CONFIGURATION AERODYNAMICS ANALYSIS

Vol. 1. Theory and Validations

Prepared by:
A. J. Baker, G. S. Iannelli, P.D. Manhardt and J. A. Orzechowski
COMPUTATIONAL MECHANICS CORPORATION

Prepared for:
NASA Ames Research Center
Moffett Field, California

FINAL REPORT FOR SBIR PHASE II
CONTRACT NO.: NAS2-12568

Report: CMC TR2.1 - 94
December 1993

COMPUTATIONAL MECHANICS CORPORATION
601 Concord Street, Suite 116
Knoxville, TN 37919-3382 U.S.A.

Phone: (615) 546-3664
FAX: (615) 546.7463
email: ajbaker@comco3.akcess.com

(NASA-CR-195918) AN ARBITRARY GRID CFD ALGORITHM FOR CONFIGURATION AERODYNAMICS ANALYSIS. VOLUME 1: THEORY AND VALIDATIONS Final Report (Computational Mechanics Co.) 179 p

N94-36914

Unclas

G3/61 0012081

Table of Contents

	Abstract	*iv*
	Nomenclature	*vi*
	Figures and Tables	ix
		Page
1.	**INTRODUCTION**	1
2.	**THE AERODYNAMICS PROBLEM STATEMENT**	5
	2.1 Synopsis	5
	2.2 Conservation law systems	5
	2.3 Turbulence, Reynolds-averaging	8
	2.4 Non-dimensionalization	9
	2.5 Canonical form	10
	2.6 Well-posed boundary conditions	14
3.	**APPROXIMATION, ERROR CONSTRAINT**	15
	3.1 Overview	15
	3.2 Approximation, measure of error	15
	3.3 Error extremization, the weak statement	16
	3.4 Spatial semi-discretization, finite volume, finite element	17
	3.5 Fully discrete form, algebraic statement	21
	3.6 Summary	23
4.	**WELL-POSEDNESS, STABILITY, CONVERGENCE**	25
	4.1 Overview	25
	4.2 Well-posedness, boundary conditions	25
	4.3 Stability, Taylor dissipation	29
	4.4 Accuracy, asymptotic convergence	32
	4.5 Stability, artificial dissipation	33
	4.6 Summary	39
5.	**THE *REMI* AERO CFD ALGORITHM**	43
	5.1 Synopsis	43
	5.2 Finite element TWS^h algorithm nomenclature	43
	5.3 *REMI* algorithm matrix statement illustrations	51

		Page
	5.4 The *REMI* RaNS/E algorithm	56
	5.5 Summary	64
6.	**AUXILIARY PROCEDURES, LINEAR ALGEBRA**	65
	6.1 Synopsis	65
6.2	Initial condition generation	65
6.3	Implicit Runge-Kutta algorithm	67
6.4	Equilibrium reacting air algorithm	70
6.5	Tensor matrix product factorization	71
6.6	Summary	77
7.	**DISCUSSION AND RESULTS**	79
7.1	Synopsis	79
7.2	Subsonic inviscid verifications, d=2,3	79
7.3	Transonic inviscid verifications, benchmarks, d=2	88
7.4	Supersonic inviscid verifications, d=2	93
7.5	Hypersonic Euler verification, validation, d=2 axisymmetric	100
7.6	Viscous transonic benchmark, validation, d=2	104
8.	**SUMMARY AND CONCLUSIONS**	111
	References	117
	Appendices	
	A. *AKCESS*.AERO *REMI* template, d=2, Newton	122
	B. TWS^h FE *REMI* algorithm, d=1,2,3	133
	C. *AKCESS*.AERO template, d=2, TP quasi-Newton	137
	D. *AKCESS*.AERO *REMI* template, quasi-Newton, d=3	151

Abstract

The solicitation for the SBIR Phase I project requested design and critical evaluation of a CFD algorithm applicable to three-dimensional configuration aerodynamics analysis, using an "arbitrary grid not requiring a well-ordered, body-fitted coordinate system for robustness." The Phase I project completion contributed to derivation of a "Taylor weak statement (TWS)" CFD algorithm, applicable to unsteady transonic potential, Euler and Reynolds-averaged Navier-Stokes conservation law systems, possessing the following attributes:

- completion of all theoretical details in the continuum employing calculus and vector field theory
- intrinsic embedding of sixteen previously published numerical dissipation methodologies for shock capturing and stability
- a continuum Galerkin weak statement extremizing conservation law approximate solution error for *any* approximation specification
- amenable to *any* (finite volume, finite element) spatial semi-discretization
- useable with *any* time discretization, implicit, explicit or multi-step
- a fully discrete theory algebraic system eligible for any appropriate quasi-Newton iteration method using sparse, block-banded and/or stationary relaxation solvers

The Phase I project results provided the theoretical foundation for selection of specific options for coding and verification in a Phase II project. The level for each theoretically independent option was selected as follows:

- Euler, laminar flow and Reynolds-averaged Navier-Stokes conservation law systems
- a diagonal scalar simplification to the TWS numerical dissipation construction
- well-posed boundary conditions for continuum Taylor weak statement Euler/Navier-Stokes constructions using Lyapunov stability methodology

[redacted]

- a finite element spatial semi-discretization using linear tensor product quadrilateral (hence hexahedral) elements of potentially arbitrary distortion

- single step θ-implicit and a B-stable, two-stage implicit Runge-Kutta time discretizations

- a block-banded, mesh-sweeping quasi-Newton iteration algorithm via matrix tensor products

- algebraic, block mesh strategy for verification, benchmark and validation problems, extensible to prototype three-dimensional aerodynamics geometries.

The original proposed scope of the Phase II project anticipated coding and verification for turbulent and three-dimensional flows about a generic fighter and a lex-delta configuration. As the project evolved, it became very apparent that achieving this goal from the given starting point, in the two-year performance period, was an impossibility. The project was therefore formally continued for two additional years, at no added cost, to provide time needed to thrash out the myriad details associated with the new algorithm. In this extension, a two-dimensional (only) research code (FEMNAS) was developed to confirm necessary operational details and to establish verification, benchmark and validation results. The restriction to two-dimensions was a practical necessity, especially since the tensor product quasi-Newton jacobian proved difficult to accurately develop for arbitrarily-distorted meshings.

The results of these fundamental 2-D simulations for transonic, supersonic and hypersonic inviscid and laminar-viscous flow test cases constitutes the major verifications reported herein. Additionally, the 3-D algorithm and associated quasi-Newton tensor product jacobian are fully presented herein, along with very basic verification and benchmark tests results. The 3-D theory is only now approaching operational readiness in the production *AKCESS.** code, which is briefly described and detailed.

NOMENCLATURE

a	scalar convection speed,
A	element matrix prefix, d=1
A_k	kinetic flux vector jacobian
B	element matrix prefix, d=2
C	Courant number, element matrix prefix, d=3
$c_p c_v$	specific heat capacities
d	dimension of problem domain
D	diffusion matrix
e	mass specific total energy, element data
e^h	semi-discrete approximation error
E	volume specific total energy
Ec	Eckert number
Eu	Euler number
f_j	kinetic flux vector (resolution)
f_j^v	dissipative flux vector (resolution)
g	amplification factor
GWS	Galerkin weak statement
h	mesh measure, enthalpy
H^p	Sobolev function space
k	thermal conductivity, turbulent kinetic energy, trial space basis polynomial degree
k^t	turbulent eddy conductivity
l_D	dissipation length
l_m	mixing length
L	reference length scale
$L(\cdot)$	partial differential equation
m	mesh measure function
m_i	momentum vector (resolution)
M	mass matrix, collision factor, molecular mass
Mk	Mach reference number
n	normal coordinate
N	finite element basis function
p	pressure
Pe	Peclet number

Pr	Prandtl number
R	polytropic gas law constant
R	universal gas law constant
RQ	residual weak statement
Re	Reynolds number
s	source term
St	Stanton number
t	time
T	temperature
u_i	velocity vector (resolution)
U	reference velocity
v	scalar speed
V	convection matrix
WS	weak statement
W_i	expansion coefficient set
x_i	coordinate system (resolution)
y	wall normal coordinate
Y_i	species mass fraction
$-\rho\overline{u_i u_j}$	Reynolds stress tensor
$\overline{u_i u_j}$	kinematic Reynolds stress tensor

α	artificial dissipation parameter
β	artificial dissipation parameter
Δ	central difference first derivative operator
δ_{ij}	Kronecker delta
δ^2	central difference second derivative operator
∇^2	laplacian operator
∇^h	discrete divergence operator
ε	error, dissipation level
ε_{ijk}	alternator tensor
θ	implicitness parameter
μ	absolute viscosity
λ	Fourier mode wavelength
v	kinematic viscosity, artificial dissipation parameter
v^t	eddy viscosity
ρ	density
Φ	potential function
Φ_i	test space function set
Ψ_i	trial space function set
Ω	domain
$\partial\Omega$	boundary
ω	vorticity, wave number
$\{\ \}$	column matrix
$\{\ \}^T$	row matrix
$[\ ]$	square matrix
$\lceil\rfloor$	diagonal matrix

List of Figures

Figure		Page
3.1	Domain partitioning into a discretization, a) hexahedra on R^3, b) composite hexahedra with eight nodes and subdivision into five tetrahedra	20
3.2	Tensor product finite element domains and node coordinate dispositions, a) two-dimensional, b) three-dimensional	20
4.1	Amplification and phase velocity error distributions, various weak statement algorithms, from Chaffin and Baker(1994)	41
5.1	Finite element domains in physical space (Ω_e) and transform space ($\hat{\Omega}_e$) for tensor product basis form	46
5.2	Gauss symmetric quadrature coordinates for d=2	47
5.3	*AKCESS*.AERO *REMI* template for $\{FR\}_e$	54
5.4	*AKCESS*.AERO *REMI* template for $\{FM1\}_e$	57
6.1	*AKCESS*.AERO *REMI* template jacobian $[RE,E]_e$, d=2	76
6.2	*AKCESS*.AERO *REMI* template, TP jacobian [RE, ETP], d=2	77
7.1	Converging subsonic duct verification, a) d=2, 2:1 area ratio, b) d=3, 4:1 area ratio	80
7.2	*AKCESS*.* template for *REMI* d=2 IC generation, a) nodal density, element-averaged metric data, b) average density, Gauss quadrature element matrices.	81
7.3	*AKCESS*.* template for *REMI* d=3 IC generation, averaged density, Gauss quadrature element matrices.	82
7.4	Converging duct validation check case, d=2, a) modestly non-cartesian mesh A, b) highly distorted mesh, B.	84
7.5a	IC algorithm TWS^h density solution, d=2 converging duct, $Ma_{in} \cong 0.2$, averaged metric template, a) mesh A, b) mesh B	85

7.5b IC algorithm TWS^h density solution, d=2 converging duct, $Ma_{in} \cong 0.2$, Gauss quadrature template, c) mesh A, d) mesh B. 86

7.6 *REMI* FE TWS^h algorithm Euler steady-state pressure solutions, converging duct, $Ma_{in} \cong 0.2$, scalar β, Mesh B, d=2, a) β=0.3, b) β=0.2 87

7.7 Total pressure loss error for *REMI* Euler solutions, converging duct, $Ma_{in} \cong 0.2$, scalar β d=2, mesh B, d=2, a) β=0.3, b) β=0.2. 89

7.8 Converging duct verification, d=3, $Ma_{in} \cong 0.2$, 4:1 area ratio, a) IC density solution, b) *REMI* pressure solution with velocity overlay, scalar β=0.3. 90

7.9 deLaval nozzle verification problem, a) cross-section distributions, b) steady *REMI* nodal solution for axial momentum. 91

7.10 deLaval nozzle, unsteady *REMI* TWS^h solution for Mach number, IRK ODE algorithm, Δt=*0.005*, a) t=0.4, b) t=1.0, c) t=1.2, d) d=1.8, e) t=*2.8* 92

7.11 *REMI* algorithm TWS^h Euler solution, steady-state, 15% parabolic arc, scalar β=0.2 $\{1\}^T$ a) 65x35 mesh, b) velocity vector field, c) entropy, d) Mach number, e) axial momentum. 94

7.12 *REMI* algorithm TWS^h Euler solution, steady-state, 15% parabolic arc, Ma_{in} = 0.68, scalar β=0.2, $\{1\}^T$ perspective and contour graphs of a) axial momentum, b) transverse momentum, c) Mach number, d) pressure. 95

7.13 *REMI* algorithm TWS^h Euler solution, supersonic wedge flow, $\theta = 20^o$, β = 0.3 $\{1\}^T$, a) initial 65x35 uniform mesh, b) density isoclines, c) 1st adapted mesh, d) resultant density isoclines. 96

7.14 *REMI* algorithm TWS^h Euler solution, supersonic wedge flow, $\theta=20^o$, β = 0.3 $\{1\}^T$, a) final adapted mesh, b) density isoclines; contour and perspective graphs, c) Mach number, d) pressure. 98

7.15 *REMI* algorithm TWS^h Euler solution, supersonic shock reflection, β = 0.3 $\{1\}^T$, solution-adapted 65x35 meshing, resultant density isocline distributions. 99

7.16 *REMI* algorithm TWS^h Euler solutions, supersonic shock reflection, $\beta_a = 0.3\ \{1\}^T$, final mesh, contour and perspective graphs, a) pressure, b) entropy. 101

7.17 *REMI* algorithm TWS^h steady state Euler solutions, hypersonic blunt-body flow adapted 65x35 quad meshes, Ma_∞=6.5, a) mesh, b) density distribution; Ma_∞ = 8.0, contour and perspective surface distributions of c) Mach number, d) density. 102

7.18 *REMI* algorithm TWS^h Euler solution, steady-state, Ma_∞=8, a) ideal-air and real-air stagnation streamline/body surface distributions of temperature, b) companion real-air species mass fractions 103

7.19 *REMI* algorithm TWS^h Navier-Stokes, laminar, viscous, 4% parabolic arc, Re=4.0x10^6, β=0.2$\{1\}^T$, a) non-uniform mesh, b) axial momentum, c) axial momentum plotted in nodal space. 105

7.20 *REMI* algorithm TWS^h Navier-Stokes solution laminar, 4% parabolic arc, Re=4.0x10^6, β=0.2, perspective presentations of a) axial momentum, b) pressure, c) pressure closeup near trailing edge. 106

7.21 Shock-laminar boundary layer validation problem. 107

7.22 *REMI* algorithm TWS^h Navier-Stokes solution, shock laminar boundary layer, Re=10_5, β = 0.3,$\{1\}^T$ a) density, b) Mach number, c) axial momentum. 108

7.23 Supersonic shock-boundary layer interaction, Ma_{in} = 2.15, Re=10^5, a) *REMI* separation region velocity resolution; comparisons on b) surface pressures c) skin friction, symbols are data from Degrez *et al* (1987) 109

8.1 Sub-grid *p*-embedding FE verification problems, inviscid square wave, a) standard WS^h solution, *k*=1 or 2, b) *p*-embedded solution, *k*=1; viscous Burgers shoch simulation, Re=10^5, c) standard WS^h, *k*=1 or 2, solution, converged to 10^3, d) *p*-embedded solution, *k*=1, converged to 10^{-9}. 116

List of Tables

Table		Page
2.1	Euler-admissable Dirichlet boundary conditions (BC)	14
4.1	Summary of CFD algorithms within Taylor weak statement, from Baker and Kim (1987)	40
5.1	Gauss quadrature coordinates and weights, d=2	47
5.2	FE k=1 basis interpolation matrix [M200] for d=1,2,3	48

I. INTRODUCTION

The work plan of this SBIR Phase II contractual project called for derivation, coding and broad-range validation of a new finite element CFD algorithm, stable, accurate and efficient on absolutely arbitrary meshings. The focus was on application to configuration aerodynamics analyses for general three-dimensional geometries using inviscid flow (Euler), laminar, and Reynolds-averaged Navier-Stokes (RaNS) simulations. Adjunct to this was generation of discretizations of three-dimensional regions, bounded by aerodynamics surfaces, with meshing versatility commensurate with the capabilities of a CFD algorithm itself.

In the past two decades the CFD community has witnessed an exceptional expenditure of resources, both technical personnel and computer, applied to the configuration aerodynamics CFD analysis requirement. The historic Euler aerodynamics development was the MacCormack (1969) explicit CFD algorithm, which originally appeared inappropriate for RaNS applications due to meshing-induced parasitic stiffness. (It has since enjoyed a rebirth, due to its total vectorizability.) Several research projects were thereby initiated in the middle 1970s, leading to development of the Beam-Warming (1976) implicit factored algorithm, designed for efficient iteration with relative insensitivity to parasitic stiffness.

Both the MacCormack and Beam-Warming CFD theories employed added artificial diffusion, viewed as both a theoretical detraction and a potential compromise to genuine viscous aerodynamics analysis. This prompted development of flux vector splitting methods for hyperbolic conservation law systems, with extension to RaNS via dissipative flux-vector central differencing. Many variants were developed, *e.g.*, Steger-Warming (1978), vanLeer (1982), Osher (1978), Roe (1981), all employing some combination of Riemann solvers, averaged-states and directional upwind differencing. The common distinguishing feature was absence of specifically-added artificial diffusion. However, numerical diffusion was intrinsic to characteristic-direction differencing, and a variety of switches for stencil expansion were developed to avoid the low order accuracy of direct upwind differencing.

These various CFD algorithms were code-implemented using finite difference or finite volume spatial semi-discretizations. The abiding character was use of structured, nominally-uniform cartesian meshings on the transformed computational domain, following a boundary-fitted coordinate transformation from the transformed aerodynamics geometry. An alternative spatial semi-discretization procedure emerged

in the 1980's, using potentially "absolutely non-structured meshings" of finite elements in the physical domain, *e.g.*, Loehner, *et al* (1984), Oden, *et al* (1986). The application codes were time explicit, generally restricted to Euler simulations and employed specifically added numerical dissipation derived from the Taylor Galerkin generalization (Donea 1984) of the Lax-Wendroff method. Locally refined applications using nested triangles and tetrahedra were demonstrated, and meshing and FE basis (termed "*hp*") solution adaptive methodology appeared promising for enhanced accuracy with degree-of-freedom efficiency via local error estimation.

This bright promise is moderated by associated computational issues such as large memory requirements, parasitic stiffness, mesh generation/adaptation procedures and extra solution steps to compute error measure data for mesh refinement/de-refinement. Without exception the extension to RaNS applications has moved to structured cartesian mesh embedding in boundary layer regions, at least, and use of operator-splitting implicit time integration methods to handle parasitic stiffness. Coincidentally, triangle/tetrahedra meshings have since become amenable to finite volume CFD constructions, c.f., Barth, *et al* (1991).

From this view in 1987, the Phase II project sought to derive, code and validate the ingredients of a CFD algorithm that exhibited arbitrary mesh versatility and solution adaptability with mathematical robustness, quality (Euler) shock-capturing, and direct extension to (meshing requirements for) genuine RaNS applications. The decisions hopefully leading to attainment of this goal were:

- weak statement for extremization of approximation error
- Taylor-series Euler/RaNS flux vector manipulation to produce continuum conservation law systems with intrinsic dissipation
- well-posed boundary conditions, suitable for Euler and RaNS, enforceable via weak statement generated surface integrals
- ideal and real-gas equations of state
- θ-implicit one step and implicit two step Runge-Kutta time discretizations
- linear basis, tensor product finite element spatial semi-discretization of quads/hexahedra
- matrix tensor product derived quasi-Newton block-banded linear algebra iteration
- algebraic, block meshing amenable to solution adaptivity

We considered the above combination, if successful, would lead to attainment of the goal, *i.e.*, an arbitrary mesh, configuration aerodynamics CFD algorithm, applicable

to both Euler and RaNS conservation law descriptions. In our view, these selections circumvented some detractions of previous constructions, while exhibiting potential for mathematical robustness, code operating efficiency and use of "arbitrary" meshings.

However, the volume of detail required to achieve the verification capability of the goal was severely under-estimated, to the extent that only a fledgling, two-dimensional verification/benchmark capability on rather regular meshings accrued to the project-end after a period spanning two years. Practical factors contributing to this slow progress included delayed delivery of our SGI Model 3300 workstation (6 months) and practical difficulties in remote operation on the NASA Ames central computer (before NASNET and T-1 communication speed were available). Theoretical and code practice factors also played a central contributing role, as the elegantly clean theory proved incomplete in the key areas of numerical dissipation and tensor product matrix algebra procedures on "arbitrary" (highly distorted) meshings.

These several issues contributed to the need to request a no-cost extension over additional years. During this period, a dedicated theoretical effort pursued by Iannelli, as he developed his dissertation (1991), lead to resolution of many algorithmic issues, both theoretical and practical. Benchmarks and validations were executed by his developed FEMNAS computer code, limited to two-dimensional form as a practical necessity. The range of transonic, supersonic and hypersonic Euler and laminar Navier-Stokes problem statements for benchmark and validations reported herein were generated by this code.

The reported CFD results constitute a comprehensive, positive assessment of the developed arbitrary-grid, finite element weak statement CFD *REMI* algorithm. Although limited to two-dimensions, high quality essentially non-oscillatory (ENO) approximation to shocks in transonic, supersonic and hypersonic flows is achieved, using block solution-adaptive remeshing with a simplified diagonal form for the β-term of the underlying TWS^h theory. The resultant meshings are highly non-rectangular and all subsonic and supersonic far-field flows leave the computational domain without oscillation, via appropriate evaluations on the weak statement-generated surface integrals. The implicit Runge-Kutta time-marching algorithm appears a truly viable, stiffly-stable and second order accurate replacement for the θ-implicit single step Euler family. It is useful for shock capturing as well as handling the parasitic stiffness generated via boundary layer meshing with transverse resolution on the order of inverse Reynolds number. The range of these results are discussed following derivation and definition of the developed arbitrary grid finite element CFD algorithm.

As these advances became achieved, the three-dimensional algorithm has become operational in our emerging *AKCESS.** software platform, the successor "code" to previously established "research" codes. The resultant 3-D TWS^h CFD "*REMI*" FE algorithm is thoroughly detailed herein, including the 3-D tensor product factorized quasi-Newton jacobian. Only modest verification-level 3-D Euler numerical results generated by *AKCESS.** to date are available and included. However, we expect to move rapidly to recovery of reported FEMNAS tests, as well as benchmark extensions to 3-D, as *AKCESS.** moves to operational readiness in its parallel-processing implementation.

The near-term emergence of this versatile software platform, specifically designed to greatly shorten the time to implement/validate theoretical and/or practical musings, we hope warrants the significant government and personal resources committed to project completion. We have certainly learned a lasting lesson on estimating the effort required to convert CFD theory to genuine, robust and convergent code practice.

2. THE AERODYNAMICS PROBLEM STATEMENT

2.1 Synopsis

The goal is to establish a robust, accurate and efficient CFD algorithm for Euler and Navier-Stokes conservation law systems describing the flow state in a configuration aerodynamics problem statement. This section establishes the associated mathematical descriptions, including closure model statements for thermodynamics and dissipation mechanisms. Following nondimensionalization, an eigenvalue analysis leads to mathematical characterization of the developed conservation law systems.

2.2 Conservation law systems

The basic assumption is that newtonian conservation statements for mass, momentum and energy, coupled with closure expressions for thermodynamics, and dissipative and modeled-turbulence mechanisms, yields the desired mathematical description. Denote the created set of dependent variables, usually called the *state variable*, as $q=q(\mathbf{x},t)$, where "q" is understood to denote an array. Then, the Euler/Navier-Stokes conservation law system, i.e., the governing non-linear, partial differential equation (PDE) set, is familiarly expressed as

$$L(q)=\frac{\partial q}{\partial t}+\frac{\partial}{\partial x_j}\left(f_j-f_j^v\right)-s=0 \tag{2.1}$$

In (2.1), and in the following, $L(\cdot)$ denotes the homogeneous form of a differential equation, x_j is a scalar resolution of the (global) $\mathbf{x}$ coordinate system spanning a region Ω of d-dimensional Euclidean space $\Re^d$ ($1\leq d \leq 3$), and t denotes time. Therefore, the domain of definition of (2.1) is $R^+ \times \Omega$ with R^+ the positive real number field spanned by t and $\Omega\subset \Re^d \in (\mathbf{x},|\mathbf{x}|<\infty)$.

For aerodynamics, the state variable is usually selected as $q(\mathbf{x},t)\equiv\{\rho,\mathbf{m},E\}^T$, where $\{\cdot\}$ denotes a column array (matrix) and superscript "T" denotes its transpose. Therein, ρ is fluid density, $\mathbf{m}$ is the momentum vector with scalar resolution $m_i = \rho u_i$, where u_i is termed "*velocity*," and $E=\rho e$ is the volume specific total internal energy.

Continuing in (2.1), f_j is the scalar resolution of the *kinetic flux vector* $\mathbf{f}$, while f_j^v is the corresponding resolution of the *dissipative flux vector* $\mathbf{f}^v$. Finally, s is a source term array, included for generality as needed, dependent on closure modeling. Both flux vectors contain pressure, and the corresponding *equation of state* functional form is $p=p(q)=p(\rho,\mathbf{m},E)$.

For the Euler and/or Navier-Stokes description, the cartesian tensor form for the terms in (2.1), for laminar viscous flow with heat transfer, is

$$q \equiv \begin{Bmatrix} \rho \\ m_i \\ E \end{Bmatrix}, \quad \mathbf{f} \Rightarrow f_j = \begin{Bmatrix} m_j \\ m_i m_j / \rho + p\delta_{ij} \\ (E+p)m_j / \rho \end{Bmatrix}, \quad \mathbf{f}^v \Rightarrow f_j^v = \begin{Bmatrix} 0 \\ \sigma_{ij} \\ \sigma_{ij} m_i / \rho - q_j \end{Bmatrix} \tag{2.2}$$

Note that "i" is a free index in the second lines in (2.2), with range $1 \le i \le d$, where d is the problem statement dimensionality. The state variable form for the polytropic perfect gas equation of state is

$$p = (\gamma - 1)(E - m_i m_i / 2\rho) \tag{2.3}$$

where γ is the ratio of specific heats. The Stokes stress tensor and Fourier heat flux vector definitions are

$$\sigma_{ij} = \mu(T)\left(\frac{\partial(m_i/\rho)}{\partial x_j} + \frac{\partial(m_j/\rho)}{\partial x_i} - \frac{2}{3}\frac{\partial(m_k/\rho)}{\partial x_k}\delta_{ij} \right) \tag{2.4}$$

$$q_j = -k(T)\left(\frac{\partial T}{\partial x_j} \right) \tag{2.5}$$

where μ is the fluid absolute viscosity, k is the fluid thermal conductivity, and both depend on the temperature $T = T(\rho, p)$.

Ultimately, pressure, hence the equation of state, plays a central role in algorithm construction and in boundary condition well-posedness for (2.1). The familiar polytropic perfect gas law form is

$$p = \rho R T \tag{2.6}$$

where R is the gas law constant (equal to the universal constant divided by molecular mass). The formulational connections between p and q lead to the following useful equations for a perfect gas

$$\begin{aligned} E &\equiv \rho e = \rho\varepsilon + m_i m_i / 2\rho \\ \varepsilon &\equiv c_v T = RT/(\gamma - 1) \\ p &= (\gamma - 1)\rho\varepsilon \end{aligned} \tag{2.7}$$

In (2.7), ε is the mass specific internal energy (usually denoted "u" in thermodynamics texts), and $\gamma \equiv c_p / c_v$ is the specific heat ratio.

The aerodynamics class of long-term interest extends to inclusion of real-gas effects, as occurs at hypersonic flight Mach number in the atmosphere. An expository formulation is a five species reacting air model, whereupon the Dalton's law generalization of (2.6) is

$$p = R\rho T \sum_{i=1}^{5} \frac{Y_i}{M_i} \tag{2.8}$$

where R (not italicized) is the universal gas constant, M_i is the molecular mass of the ith species, and Y_i is the associated mass fraction. For species ordering (O, N, NO, O_2, N_2), for $1 \leq i \leq 5$, the replacement expression for specific internal energy (2.7) is

$$\varepsilon = -RT\sum_{i=1}^{5}\frac{Y_i}{M_i} + \sum_{i=1}^{5} Y_i c_{pi} T + \sum_{i=3}^{5} Y_i \frac{R\theta_i^v / M_i}{\exp\left(\theta_i^v / T\right) - 1} + \sum_{i=1}^{3} Y_i h_i^0 \tag{2.9}$$

where $c_{pi} = 5/2(R/M_i), i = 1,2$, $c_{pi} = 5/2(R/M_i)$ $, i = 1,2$, and $c_{pi} = 7/2(R/M_i), i = 3,4,5$. Further, in (2.9) h_i^o is the formation enthalpy and θ_i^v is the characteristic vibrational temperature of species i. Thus, ε is now comprised of species formation energy, the molecular translational and rotational kinetic energies, and the molecular vibrational potential energy, with equilibrium magnitude characterized by static temperature T.

The exchange reactions for the five species air model are

$$\begin{aligned} O_2 + M &\leftrightarrow 2O + M \\ N_2 + M &\leftrightarrow 2N + M \\ N_2 + O_2 + M &\leftrightarrow 2NO + M \end{aligned} \tag{2.10}$$

where M denotes a collision factor. The law of mass action (Anderson, 1989) applied to (2.10) requires

$$\frac{Y_2^2}{Y_5} = \frac{M_2^2 K_1(T)}{\rho M_5}, \quad \frac{Y_1^2}{Y_4} = \frac{M_1^2 K_2(T)}{\rho M_4}, \quad \frac{Y_3^2}{Y_4 Y_5} = \frac{M_3^2 K_3(T)}{M_4 M_5} \tag{2.11}$$

where $K_1(T)$, $K_2(T)$ and $K_3(T)$ are the available equilibrium reaction relations (c.f., Park, 1989). The mass action relations (2.11) constitute three constraints on the five mass fractions Y_i. The additional constraints are global mass conservation

$$\sum_{i=1}^{5} Y_i = 1 \tag{2.12}$$

and uniformity of the proportion of oxygen to nitrogen nuclei for air, i.e.

$$\left(\frac{Y_1}{M_1} + \frac{Y_3}{M_3} + 2\frac{Y_4}{M_4}\right)/21 = \left(\frac{Y_2}{M_2} + \frac{Y_3}{M_3} + 2\frac{Y_5}{M_5}\right)/79 \tag{2.13}$$

The algorithmic procedure developed to solve (2.8)-(2.13) is detailed in a latter section.

The final step to closure of (2.1)-(2.5) is definition of the functional relationship for $\mu(T)$ and $k(T)$ in (2.4)-(2.5). For the Euler description, these data are identically zero. Otherwise, referenced to the standard atmosphere at T_r = 273.1K (529R), the Sutherland's correlations are (White, 1978)

$$\frac{\mu}{\mu_r} = \left(\frac{T}{T_r}\right)^{3/2}\left(\frac{1+\omega}{T/T_r+\omega}\right)$$
$$\frac{k}{k_r} = \left(\frac{T}{T_r}\right)^{3/2}\left(\frac{1+\phi}{T/T_r+\phi}\right) \tag{2.14}$$

where μ_r and k_r are data at T_r and for air, $\omega \equiv 0.4048$ and $\phi \equiv 0.7120$ in SI units.

2.3. Turbulence, Reynolds-averaging

At flight conditions, the presented Navier-Stokes description requires a manipulation, commonly termed "*Reynolds-averaging,*" to establish an aerodynamics conservation law statement amenable to computing for turbulent flows. (The exception is when one uses a "direct numerical simulation (DNS)" procedure, cf., Moin(1992), which seeks highly time-accurate evolution of a geometrically-elementary model description.) Numerous Reynolds-averaging procedures are available, ranging from sub-grid scale (SGS) modeling to time- or volume-based single point correlations. Renormalization group theory (RNG) has emerged recently, Yakhot, *et al*, (1986), as a replacement theory for establishing the range of historical Reynolds-averaged constructions and closure models, as well for deriving new formulations.

With brevity, the end-point of a Reynolds-averaging procedure replaces the state variable entries with the corresponding "mean" variables $\{\bar{\rho}, \bar{\mathbf{m}}, \bar{E}\}^T$, with corresponding kinetic flux vector variable notation. The dissipative flux vector becomes augmented to the form

$$f_j^v = \begin{Bmatrix} 0 \\ \overline{\sigma_{ij}} - \overline{\rho u_i u_j} \\ \overline{\sigma_{ij}} - \overline{\rho u_i u_j})\overline{m_i}/\bar{\rho} - \bar{q}_j - \overline{h u_j} \end{Bmatrix} \tag{2.15}$$

In (2.15) a superscript overbar denotes a "Reynolds-average" of the indicated variable product, and the enthalpy definition is $h=(e+p/\rho)$. The new unknowns introduced into (2.2) via (2.15) are termed the "Reynolds stress tensor" $\overline{\rho u_i u_j}$ and the turbulent heat flux vector $\overline{h u_j}$, where u_i remains the velocity, i.e., the ratio m_i/ρ .

It is well beyond the scope of this project to pursue use or validation of a turbulence closure model for Reynolds stress tensor or turbulent heat flux vector. For formulation completeness, however, the essence of the simplest "eddy viscosity" closure, e.g., the Baldwin-Lomax (1975) model, is summarized by modifications to (2.4)-(2.6) to the forms

$$\sigma_{ij} = \overline{\sigma_{ij}} - \overline{\rho u_i u_j} = \left(\mu(\overline{T}) + \mu^t\right)\overline{E}ij \tag{2.16}$$

$$q_j = \overline{q}_j - \overline{hu}j = -\left(k(\overline{T}) + k^t\right)\frac{\partial \overline{T}}{\partial x_j} \tag{2.17}$$

$$p = \overline{p} + \overline{\rho u_k u_k} = \overline{\rho}(R\overline{T} + 2k/3) \tag{2.18}$$

The new variables thereby introduced are μ^t, the "turbulent viscosity," k^t , the "turbulent thermal conductivity," and one-half the trace k of the *kinematic* Reynolds stress tensor $\overline{u_i u_j}$ termed "turbulent kinetic energy," Further $\overline{E}_{ij}$ in (2.16) is the mean flow deviatoric strain rate tensor, which is the parenthetical expression in (2.4) expressed in mean (Reynolds-averaged) variables.

The eddy viscosity closure model thus reduces the new unknowns to the scalars μ^t and k^t , each of which is a function of the entire mean flow state variable q rather than (only) the thermodynamic state. Importantly, (2.1) remains the conservation law form for all descriptions pertinent to configuration aerodynamics problems.

2.4 Non-dimensionalization

The terminal preparation step is selecting a suitable reference state, such that (2.1)-(2.3), conceptually augmented with (2.15), can be identified for essential character. Several non-dimensionalization (non-D) forms are familiar, dependent upon what is selected as the reference primitives. The general form results by selecting a reference state for each member of the state variable $q = \{\rho, \mathbf{m}, E\}^T$ and the pressure p. Hence, denoting a non-D variable by an asterisk, and the reference by a subscript "r," define

$$\begin{aligned} \rho^* &= \rho/\rho_r & \mathbf{m}^* &= \mathbf{m}/m_r = \rho\mathbf{u}/\rho_r U_r \\ E^* &= E/E_r & p^* &= p/p_r \\ \mathbf{x}^* &= \mathbf{x}/L_r & t^* &= t/t_r \equiv t/(L/U_r) \end{aligned} \tag{2.19}$$

Substituting (2.19) into (2.1)-(2.3) and clearing produces the following non-D groups:

$$\begin{aligned} &\text{Reynolds number: } \mathrm{Re} = \rho_r U_r L/\mu_r & &\text{Euler number: } \mathrm{Eu} = p_r/\rho_r U_r^2 \\ &\text{Prandtl number: } \mathrm{Pr} = c_{pr}\mu_r/k_r & &\text{Mach reference: } \mathrm{Mk} = U_r/\sqrt{\gamma p_r/\rho_r} \\ &\text{Eckert number: } \mathrm{Ec} \equiv E_r/\rho_r U_r^2 & & \\ &\qquad = \left[\frac{1}{2} + \frac{1}{\gamma(\gamma-1)\mathrm{Mk}^2}\right] & &\text{Stanton number: } \mathrm{St} = L/U_r t_r \equiv 1 \end{aligned} \tag{2.20}$$

With definitions (2.19), and non-D groups (2.20), the non-D form for (2.1) remains as expressed, upon replacement of (2.2) in non-dimensional variable form as

$$q=\begin{Bmatrix}\rho\\ m_i\\ E\end{Bmatrix},\quad f_j=\begin{Bmatrix}m_j\\ m_i m_j/\rho+\mathrm{Eu}p\delta_{ij}\\ (E+(\mathrm{Eu}/\mathrm{Ec})p)m_j/\rho\end{Bmatrix}\quad f_j^v=\begin{Bmatrix}0\\ \mathrm{Re}^{-1}\sigma_{ij}\\ (\mathrm{ReEc})^{-1}\sigma_{ij}m_i/\rho-(\mathrm{Pe})^{-1}q_j\end{Bmatrix}\tag{2.21}$$

The superscript asterisks are suppressed in (2.21), and the non-D functional forms for σ_{ij} and q_j remain as expressed by (2.4)-(2.5). Finally, the non-D polytropic gas equation of state replacement for (2.3) is

$$p=\frac{\gamma-1}{\mathrm{Eu}}(\mathrm{Ec}E-m_j m_j/2\rho)\tag{2.22}$$

Several special cases of this non-dimensionalization have been historically used. For example, if the reference pressure is equated to the dynamic pressure, then Eu=1 by definition. Substituting this into (2.20) then produces Mk=0.84515, Ec=$(\gamma+1)/2(\gamma-1)$=3 for air, and Pe=RePr$(\gamma+1)/2$=0.85714 RePr for air. Alternatively, if one defines Eu=γ^{-1}, then Mk=1.0 and Ec and Pe are appropriately modified. Another non-dimensionalization defines Ec=1, hence for air, Mk=$\sqrt{2/\gamma(\gamma-1)}=1.8898$, Eu = $(\gamma-1)/2=0.2$, Pe = $2\mathrm{RePr}/\gamma$ and (2.3) is identically the non-D form for pressure.

Thus, each of these special cases basically constitutes definition of a (fictitious) Mach reference, Mk, which thereby alters the scalings throughout (2.21)-(2.22). The given forms are sufficiently general to allow independent selection of Mk, which is most logically connected to the aerodynamic freestream state.

2.5 Canonical form

The non-dimensional Euler/Reynolds-averaged Navier-Stokes (E/RaNS) conservation law system is now identified as (2.1) with state variable $q=\{\rho,\mathbf{m},E\}$, appropriately interpreted, and with flux vectors (2.20)-(2.21) and pressure (2.22). Assume existence of a general coordinate transformation $x_j=x_j(\eta_k)$, where η_k is a (curvilinear) resolution that may be "boundary-fitted" for an aerodynamics description. Using the chain rule, (2.2) can be expressed as

$$\frac{\partial q}{\partial t}+\frac{e_{jk}}{\det}\frac{\partial}{\partial\eta_k}\left(f_j-f_j^v\right)-s=0\tag{2.23}$$

where e_{jk} are elements in the matrix of co-factors, constructed from the (known) forward transformation jacobian $[\partial x_j/\partial\eta_k]$, and "det" is the determinant of this matrix. (Hereon, $[\cdot]$ denotes a square, non-singular matrix while $\{\cdot\}$ remains a column matrix).

For time-invariance of the transformation, and invoking the fundamental metric invariance identity $\partial(e_{jk})/\partial\eta_k = 0$, (2.23) can be rearranged to the form

$$L(q) = \frac{\partial}{\partial t}(\det q) + \frac{\partial}{\partial \eta_k}\left(e_{jk}\left(f_j - f_j^v\right)\right) - s^* = 0 \tag{2.24}$$

where s^*=det s. In the literature, (2.23) has been termed a "*weak*" conservation form, while (2.24) is the "*strong*" form, cf., Pulliam(1986). An interesting observation accrues to completing the first products in $e_{jk}f_j$. Multiplying the first entry in (2.2) by ρ/ρ, the *convective* (only) flux vector resolution f_k^q of q becomes

$$f_k^q \equiv e_{jk} f_j^q = e_{jk}\left(\frac{m_j}{\rho}\right) q \equiv v_k q \tag{2.25}$$

where v_k is termed the (streamline) *contravariant* resolution of the convection velocity vector $\mathbf{m}/\rho$. For the η_k coordinate transformation "body-fitted," then (2.25) expresses velocity principal components near surfaces, hence the local tangent-normal resolution.

The transformation also facilitates the meaningful characterization of the mathematical form (2.23). In the limit Re grows without bound, the dissipative flux vector f^v vanishes, cf., (2.21), as does the source term, yielding the Euler form

$$L^E(q) = \frac{\partial q}{\partial t} + \frac{1}{\det}\frac{\partial}{\partial \eta_k}\left(e_{jk} f_j\right) = 0 \tag{2.26}$$

Realizing that $f_j = f_j(q)$, the *canonical form* of (2.26) is

$$\frac{\partial q}{\partial t} + \frac{1}{\det}\frac{\partial(e_{jk} f_j)}{\partial \eta_k} = 0 \tag{2.27}$$

The jacobian of the kinetic flux vector under the transformation i.e., $A_k = \partial(e_{jk}f_j)/\partial q$, is the order ($d$+2) matrix

$$A_k = \begin{bmatrix} 0 & , \; e_{\ell k} & , \; 0 \\ -e_{jk}\dfrac{m_j}{\rho^2}m_i + e_{ik}p_\rho & , -e_{\ell k}\dfrac{m_j}{\rho} + e_{\ell k}p_{m_\ell} + e_{jk}\dfrac{m_j}{\rho}\delta_{i\ell} & , \; e_{ik}p_E \\ -e_{jk}\dfrac{m_j}{\rho^2}(E+p) + e_{jk}\dfrac{m_j}{\rho}p_\rho & , \dfrac{e_{\ell k}}{\rho}(E+p) + e_{jk}\dfrac{m_j}{\rho}p_{m_\ell} & , \; e_{jk}\dfrac{m_j(1+p_E)}{\rho} \end{bmatrix} \tag{2.28}$$

Here, the indices $1 \le (i,l) \le d$ are free, corresponding to the d-dimensional resolution of the momentum vector $\mathbf{m}$. Further, the subscript notation p_ρ , p_{m_ℓ} and p_E signifies the corresponding partial derivatives of pressure with elements of the state variable q.

The mathematical character of (2.26) depends upon the eigenvalue composition, c.f., Courant and Hilbert (1929), of the matrix

$$A_\omega \equiv \sum_{k=1}^{n} \omega_k \frac{\partial\left(e_{jk} f_j\right)}{\partial q} \tag{2.29}$$

for $\sum \omega_k \equiv 1,\ 1 \le k \le n$ and all $\omega_k \in R^+$. Due to the linear composition of A_k, with respect to the e_{ik} in (2.28), the form A_ω coincides with A_k upon the substitution of e_{jk} with $e_{j\omega} = \omega_k e_{jk}$ for all j. Therefore, the eigenvalues of A_ω in (2.29) are real (or complex) if and only if those of A_k in (2.28) are correspondingly composed. The eigenvalues of the kinetic flux vector jacobian are the solution to the characteristic equation

$$\det\left[\frac{\partial\left(e_{jk} f_j(q)\right)}{\partial q} - \lambda_k I\right] = 0 \tag{2.30}$$

which is a real coefficient algebraic equation of degree $m=d+2$ with exactly m roots. The characteristic equation is thus of the form

$$\lambda_k^m - \kappa_{1_{\underline{k}}} \lambda_k^{m-1} + \kappa_{2_{\underline{k}}} \lambda_k^{m-2} + \cdots + (-1)^m \kappa_{m_{\underline{k}}} = 0 \tag{2.31}$$

with coefficients κ_i expressed in terms of the trace s_i of the ith power A_k via the recursion relations

$$\begin{aligned} &\kappa_1 = s_1,\ \ \kappa_2 = -\frac{1}{2}(s_2 - \kappa_1 s_1),\ \ \kappa_3 = \frac{1}{3}(s_3 - \kappa_1 s_2 + \kappa_2 s_1),\ \ldots \\ &\kappa_m = \frac{(-1)^{m-1}}{m}\left(s_m - \kappa_1 s_{m-1} + \kappa_2 s_{m-2} \ldots \pm \kappa_{m-1} s_1\right) \end{aligned} \tag{2.32}$$

which reduces the (formidable) operation of determinant formation in (2.30). For $d=2$, and with subscript k denoting scalar resolution, (2.32) for (2.30) becomes

$$\begin{aligned}
\kappa_{1_k} &= 4 e_{jk}\left(\frac{m_j}{\rho}\right) + e_{jk} p_{mj} + e_{jk} \frac{m_j}{\rho} p_E \\
\kappa_{2_k} &= 3\left(e_{jk}\frac{m_j}{\rho}\right)^2 + 3 e_{jk}\frac{m_j}{\rho}\left(e_{jk}\frac{m_j}{\rho}(1+p_E) + e_{jk} p_{mj}\right) \\
&\quad - e_{j_{\underline{k}}} e_{j_{\underline{k}}}\left(\frac{m_i}{\rho} p_{m_j} + p_\rho + \frac{p_E}{\rho}(E+p)\right) \\
\kappa_{3_k} &= 2\kappa_{2_k}\left(e_{jk}\frac{m_j}{\rho}\right) - 3\kappa_{1_k}\left(e_{jk}\frac{m_j}{\rho}\right)^2 + 4\left(e_{jk}\frac{m_j}{\rho}\right)^3 \\
\kappa_{4_k} &= \kappa_{3_k}\left(e_{jk}\frac{m_j}{\rho}\right) - \kappa_{2k}\left(e_{jk}\frac{m_j}{\rho}\right)^2 + \kappa_{1_k}\left(e_{jk}\frac{m_j}{\rho}\right)^3 - \left(e_{jk}\frac{m_j}{\rho}\right)^4
\end{aligned} \tag{2.33}$$

Hence, roots of (2.31) yield the eigenvalues of A_k as

$$\lambda_{k_{1,2}} = e_{jk}\frac{m_j}{\rho}$$

$$\lambda_{k_{3,4}} = e_{jk}\frac{m_j}{\rho} + \frac{1}{2}\left(e_{jk}p_{m_j} + e_{jk}\frac{m_j}{\rho}p_E\right) \tag{2.34}$$

$$\pm\left(\frac{1}{4}\left(e_{jk}p_{m_j} + e_{jk}\frac{m_j}{\rho}p_E\right)^2 + e_{j\underline{k}}e_{j\underline{k}}\left(\frac{m_j}{\rho}p_{m_j}p_\rho + \frac{p_E}{\rho}(E+p)\right)\right)^{1/2} \tag{2.35}$$

With extra labor, the d=3 equivalents of (2.37)-(2.34) can be similarly established. For d=2 (3), the first pair (triple) of eigenvalues $\lambda_{k1,2}$ ($\lambda_{k1,2,3}$) correspond to the convective velocity resolution in *contravariant* scalar components, recall (2.25). The remaining two eigenvalues will also be real provided the square root arguments are positive. A homogeneous fluid, e.g., (2.6)-(2.7), leads to a considerable simplification of (2.35). Specifically,

$$e_{jk}p_{m_j} = e_{jk}\left(\frac{\partial p}{\partial \varepsilon}\right)_\rho\left(\frac{\partial \varepsilon}{\partial m_j}\right) = -\left(\frac{\partial p}{\partial \varepsilon}\right)_\rho e_{jk}\frac{m_j}{\rho^2} \tag{2.36}$$

$$e_{jk}\frac{m_j}{\rho}p_E = e_{jk}\frac{m_j}{\rho}\left(\frac{\partial p}{\partial \varepsilon}\right)_\rho\left(\frac{\partial \varepsilon}{\partial \rho}\right) = \left(\frac{\partial p}{\partial \varepsilon}\right)_\rho e_{jk}\frac{m_j}{\rho^2} \tag{2.37}$$

hence

$$\lambda_{k_{3,4}} = e_{jk}\frac{m_j}{\rho} \pm \left(e_{j\underline{k}}e_{j\underline{k}}\left(\frac{m_j}{\rho}p_{m_j} + p_\rho + \frac{p_E}{\rho}(E+p)\right)\right)^{1/2} \tag{2.38}$$

where the underscore (on k) denotes *not* a summation index.

For a rectangular Cartesian coordinate system, the metrics e_{jk} reduce to (1,0), hence 2.38) becomes

$$\lambda_{k_{3,4}} = \frac{m_k}{\rho} \pm \left(\frac{m_j}{\rho}p_{m_j} + p_\rho + \frac{p_E}{\rho}(E+p)\right)^{1/2} \tag{2.39}$$

In this simplest geometry, it is well known that

$$\gamma_{k_{3,4}} = \frac{m_k}{\rho} \pm c \tag{2.40}$$

where c is the isentropic sound speed $\sqrt{\gamma\, p/\rho}$. Hence, (2.39) provides the auxiliary definition

$$c \equiv \sqrt{\left.\frac{\partial p}{\partial \rho}\right|_s} = \left(\frac{m_j}{\rho}p_{m_j} + p_\rho + \frac{p_E}{\rho}(E+p)\right)^{1/2} \tag{2.41}$$

and the argument in the square root in (2.39) is indeed positive. By implication then, (2.38) yields two real characteristics, hence the Euler conservation law system (2.26) is *hyperbolic*, as was known at the outset. Thereby, the eigenvalues (2.34) correspond to convective waves with characteristic contravariant celerities $v_{k\ 1,2}$ while eigenvalues (2.35) correspond to combined convective-pressure waves with characteristic celerities $v_{k3,4}$. Then, the Reynolds-averaged NS conservation law system (2.21), at large but finite Reynolds number, constitutes a "parabolic perturbation" to the hyperbolic Euler system.

2.6 Well-posed boundary conditions

The Euler form (2.1) has been thoroughly analyzed for well-posedness regarding applicable fixed (Dirichlet) boundary conditions, cf., Strickwerda(1977). Configuration aerodynamics problem statements involve boundaries upon which inflow, outflow and no throughflow (walls) occur. Further, admissible boundary constraints are characterized according to whether the flow is (locally) supersonic or subsonic. For d the problem dimension, the applicable *number* of fixed (Dirichlet) boundary conditions, admissible for well-posedness, are given in Table 2.1.

Table 2.1 Euler-admissible Dirichlet Boundary Conditions (BC)

TYPE	FLOW STATE	NO. OF DIRICHLET BC
Inflow	Supersonic	d+2
	Subsonic	d+1
Outflow	Supersonic	0
	Subsonic	1
Wall	Either	None

The hyperbolic conservation law characteristics analysis is not deterministic regarding which state variable members may be held by a Dirichlet constraint, which thus generates an analysis requirement. This is completed in Section 4.

3. APPROXIMATION, ERROR CONSTRAINT

3.1 Overview

The Euler, Navier-Stokes and Reynolds-averaged Navier-Stokes conservation law systems, with closure model essence for the configuration aerodynamics problem class, are developed including non-dimensionalization. The premise here is that the associated state variable $q(\mathbf{x},t)$, which fully describes the flow state, is not attainable in closed form. Thereby, the development of a theory to support the establishment of an *approximation* $q^N(\mathbf{x},t)$ to $q(\mathbf{x},t)$ is required.

The mathematical approach is to establish a suitable *measure* of the *error* associated with *any approximation*, and then define a process that renders this error an *extremum*. The resultant theory and procedural process employs the *weak statement*, which requires that the algorithm designer identify suitable function spaces upon which to draw to support the developed ingredients. A key attribute of this process is that it proceeds in the *continuum*, hence uses vector field theory and calculus for all processes. Upon theory completion, one then retraces the steps using a specific selection for *trial space* and *test function* sets, which herein coincides with the decision to employ a spatial semi-discretization to render the defined integrals easily evaluated. This chapter develops the weak statement for solving the aerodynamics conservation law system (2.1), (2.21)-(2.22)

3.2 Approximation, measure of error

Mathematically, *any* (CFD) algorithm seeks to generate an approximation $q^N(\mathbf{x},t)$, to the analytical state variable $q(\mathbf{x},t)$, via a denumerable set of decisions leading to an algebraic equation amenable "to computing." The mathematician's *weak statement* has now emerged, c.f., Oden and Reddy(1976), under significant "coaxing" by academic engineers, c.f., Baker(1991, 1983), to largely encompass *all* predecessor CFD theoretical procedures.

Weak statement theory suggests the starting point as selection of a space of functions that are suitable to "support" an approximation. Denoting members of this set as $\Psi_j(\mathbf{x})$, *any* approximation to the state variable $q(\mathbf{x},t)$ satisfying (2.1) is

$$q(\mathbf{x},t) \simeq q^N(\mathbf{x},t) \equiv \sum_{j=1}^{N} \Psi_j(\mathbf{x}) Q_j(t) \tag{3.1}$$

Here, superscript N denotes "approximation," and (3.1) indicates the construction as products of *known* functions $\Psi_j(\mathbf{x})$, collectively called the "*trial space*," and a set of *unknown* expansion coefficients $Q_j(t)$. Since (2.1) is initial-value, space and time are

indeed separable coordinates, hence $Q_j(t)$ can totally support all time dependence in any approximation. Since the trial space members $\Psi_j(\mathbf{x})$ are assumed known, all members therein must be specified. The available choices are essentially limitless, *e.g.*, trigonomic polynomials, Fourier series, Legendre polynomials, Chebyshev polynomials, Lagrange or Hermite interpolation polynomials, etc. Note that the *quality* of $q^N(\mathbf{x},t)$ depends in the most fundamental way on this choice.

Indeed, (3.1) is the statement of *any* approximate solution, but it contains no information on how good a specific q^N may be. Since q^N is an approximation, it cannot satisfy (2.1) identically. Specifically, $L(q^N)$ does not equate to zero, and the amount by which it is not zero is a measure of (the distribution of) the approximation error e^N

$$e^N = e^N(\mathbf{x},t) = q - q^N \tag{3.2}$$

Thus, the optimal criterion for design of a CFD algorithm must be to absolutely *minimize* this error. Since e^N is itself not known, then the available measure *extremum* occurs when the integral

$$\int_\Omega w(x,t) L(q^N) dt \equiv 0 \text{ , } \textit{for all } w(x,t) \tag{3.3}$$

vanishes for *any* "test function" $w(\mathbf{x},t)$, for which the choice is limitless.

Selecting a specific trial space set $\Psi_j(\mathbf{x})$, and for any test function $w(\mathbf{x},t)$, completing the integrals in (3.3) always produces an ordinary differential equation (ODE) system. Hence, (3.1)-(3.3) constitutes an *integral transformation* of a system of PDEs, with solution approximation $q^N(\mathbf{x},t)$, into a (much) larger system of ODEs written on $Q_j(t)$, the evolution of the solution expansion coefficient set.

Any discrete time integration method is potentially applicable to this generated ODE system, *e.g.*, Adams-Bashforth-Moulton, Runge-Kutta, Euler, trapezoidal rule, leapfrog, etc. Substituting (3.3) into the corresponding Taylor series always yields an algebraic statement, which is the terminal "computable form" of *any* CFD algorithm. This algebraic system is strongly nonlinear, hence the CFD algorithm designer faces selection of an iterative strategy to actually create the fully discrete solution for Q_j at time $t = n\Delta t$. Again, a wide variety of linear algebra choices exist.

3.3 Error extremization, the weak statement

Viewing (3.1)-(3.3), a specific algorithm is constituted at least by the choices for trial space $\Psi_j(\mathbf{x}), 1 \le j \le N$, the test function $w(\mathbf{x},t)$, the ODE integration algorithm, and a matrix iteration procedure. The concepts of trial space and extremization of error are

elegant, but require detailed development to emerge into a practical numerical recipe. First, that (3.3) must hold for *any* test function must become deterministic. Certainly, any given function can be accurately represented by an interpolation. Selecting another set of functions $\Phi_i(\mathbf{x})$ for this interpolation, every test function may be approximated as

$$w(\mathbf{x},t) \simeq w^M(\mathbf{x},t) \equiv \sum_{i=1}^{M} \Phi_i(\mathbf{x}) W_i(t) \tag{3.4}$$

In (3.4), superscript "M" denotes interpolation, and any time-dependence is again cast onto the *known* expansion coefficient set $W_i(t)$.

Once the function set $\Phi_i(\mathbf{x})$ is chosen for (3.4), a specific test function interpolation $w^M(\mathbf{x},t)$ is distinguishable from any other *only* by the corresponding coefficient set $W_i(t)$. Thereby, that (3.3) must hold for *any* test function can now be precisely enforced by requiring this integral be *stationary* with respect to the (any) set of $W_i(t)$. This *extremum*, termed a "*weak statement*," is

$$\begin{aligned} WS^N &\equiv \frac{\partial}{\partial W_i} \int_\Omega w^M(\mathbf{x},t) L\left(q^N\right) d\tau \\ &= \int_\Omega \Phi_i(\mathbf{x})\, L\left(q^N\right) d\tau = 0 \ , \ \textit{for } 1 \le i \le M \equiv N \end{aligned} \tag{3.5}$$

Neglecting some (theoretical) boundary condition details, M in (3.5) is equal to N in (3.1), such that (3.5) produces an ODE system of rank precisely equal to the N unknowns $Q_j(t)$. Equation (3.5) cleanly resolves the issue of "*for all* $w(\mathbf{x})$" in (3.3), at the expense of introducing another function set $\Phi_i(\mathbf{x})$ for a decision.

The *optimal* choice for the set $\Phi_i(\mathbf{x})$ is that it be *identical*, member by member, to the trial space $\Psi_j(\mathbf{x})$. This choice yields the *Galerkin weak statement*

$$GWS \equiv \int_\Omega \Psi_j(\mathbf{x}) L\left(q_N\right) d\tau = 0 \ \textit{for } 1 \le i \le N \tag{3.6}$$

Thus, the approximation error in any $q^N(\mathbf{x},t)$ is required to be *orthogonal* to every member of the space of functions supporting q^N, for *any* choice of trial space. There is no linearity assumption for this concept to be valid, although the rigorous mathematical proof of optimality can be established only via a linearized analysis. However, quality numerical experiments have verified optimality for model problems, and select laminar and turbulent Navier-Stokes statements, c.f., Baker (1983, Ch. 4-6).

3.4 Spatial semi-discretization, finite volume, finite element

Historically, the traditional choice for a CFD numerical algorithm construction has been finite difference methodology, or more recently, with boundary-fitted

transformations, a finite volume (FV) construction. A FV CFD algorithm, for *any* approximation (3.1), starts with the form

$$FVWS^N \equiv \int_\Omega L(q^N)d\tau \tag{3.7}$$

which obviously cannot equate to zero, as does (3.6), since $L(q^N) \neq 0$. To produce a computable form, hence resolve this issue, the $FVWS^N$ must be cast onto a spatial *semi-discretization* Ω^h of Ω. Notationally, $FVWS^N \rightarrow FVWS^h$, and substituting (2.2) for clarity, (3.7) becomes the theoretical statement

$$\begin{aligned} FVWS^h &\equiv \sum_{\Omega^h} \int_\Omega \left(\frac{\partial q^h}{\partial t} + \frac{\partial}{\partial xj}\left(f_j - f_j^v\right)^h - s \right) d\tau \\ &= \sum_{\Omega^h} \int_{\Omega^h} \left(\frac{\partial q^h}{\partial t} - s \right)^h d\tau + \sum_{\partial\Omega^h} \oint_{\partial\Omega^h} \left(f_j - f_j^{\mathbf{v}}\right)^h \hat{n}_j \, d\sigma = 0 \end{aligned} \tag{3.8}$$

The replacement of the integral over Ω in (3.7), by the sum of integrals over each sub-domain in Ω^h, and *all* individual closed boundaries in $\partial\Omega^h$, allows (3.8) to be equated to zero as indicated.

Evaluating the first integral in (3.8), on the generic FV domain in Ω^h, usually involves averaged (cell-centered) data. Conversely, every closed-surface integral requires data evaluations all around the boundary $\partial\Omega^h$ of *each* volume in Ω^h. Comparing (3.8) to the general statement (3.3), hence also (3.5), the choice $w(\mathbf{x},t) \equiv$ *constant* has obviously been made. In (3.5), the interpolation polynomial set Φ_i is thus reduced to the single constant (unity), hence a FV algorithm is a weak statement (3.5) wherein the approximation error is made orthogonal to the set of all constants. It constitutes a Galerkin weak statement only when the trial space $\Psi_j(\mathbf{x}) \rightarrow \Psi_j(\mathbf{x}^{\cdot\cdot})$ is also the set of (piece-wise) constants.

The alternative to the FV construction is to utilize a test function set $\Phi_i(\mathbf{x})$ which is indeed differentiable. The Galerkin construction employs $\Phi_i(\mathbf{x})$ identical to $\Psi_i(\mathbf{x})$, which are not constants, hence

$$
\begin{aligned}
GWS^N &\equiv \int_\Omega \Psi_i\,(x) L\left(q^N\right) dt = 0, \textit{ for all } i \\
&= \int_\Omega \Psi_i \left(\frac{\partial q^N}{\partial t} + \frac{\partial}{\partial x_j}\left(f_j - f_j^v\right)^N - s \right) dt \\
&= \int_\Omega \Psi_i \left(\frac{\partial q^N}{\partial t} - s \right) dt - \int_\Omega \frac{\partial \Psi_i}{\partial x_j}\left(f_j - f_j^v\right)^N dt + \oint_{\partial\Omega} \Psi_i \left(f_j - f_j^v\right)^N + \hat{n}_j d\sigma = \{0\}
\end{aligned}
\tag{3.9}
$$

using a Green-Gauss form of the divergence theorem. No discretization was required to manipulate (3.9), hence no restrictions underlie the choice for $\Psi_i(\mathbf{x})$ except differentiability. In further distinction to (3.8), the last term in (3.9) requires evaluation *only* on the problem domain boundary, $\partial\Omega$, hence provides the slot to *analytically* impose flux vector (embedded) boundary conditions.

The continuum GWS^N (3.9) is a function of the approximate solution trial space $\Psi_i(\mathbf{x})$, containing N members, and boundary conditions (to be identified). The integrals in (3.9) are very difficult to evaluate when the trial space set spans the entirety of Ω, i.e., is "global." The basic idea behind a *finite element* (FE) approximation to (3.9) is to re-express each $\Psi_i(\mathbf{x})$ as a set of kth-degree (interpolation) polynomials having compact support. Hence, definition of a spatial semi-discretization of Ω^h, is formed as the union (denoted "$\cup$" and meaning non-overlapping sum) of FE domains Ω_e. The geometrical shape of an Ω_e often looks identical to a FV subdomain, *e.g.*, triangles and quadrilaterals in 2D and tetrahedron and hexahedron in 3D, and the domain sides (or faces, in 3D) are straight (planar) or curved (only for FE). Figure 3.1a) illustrates a discretization of Ω^h constituted of hexahedra, and decomposition of one hexahedron into five tetrahedra, Fig. 3.1b).

The FE interpolation polynomials spanning Ω_e have *knots* (evaluation points) coinciding with geometrical distinctions, e.g., vertices, mid-edges, etc. State variable approximation expansion coefficients $Q_i(t)$ are assigned to these locations, which are then called "*nodes*" On *any* FE (or FV) region, these domain data may be clearly denoted as a column matrix $\{Q(t)\}_e$. For the FE choice, the array of kth degree interpolation polynomials on Ω_e is usually denoted $\{N_k(\eta_j)\}$, where η_j is the coordinate system intrinsic to Ω_e. Figure 3.2 illustrates curvilinear quadrilateral and hexahedral finite element domains Ω_e with vertex and mid-edge nodes denoted as ($\bullet$) and (x) respectively.

When one makes the $FVWS^h$ choice, (3.8) is usually expressed in subscript node-index notation rather than matrix notation. Conversely, an FE semi-discretization Ω^h

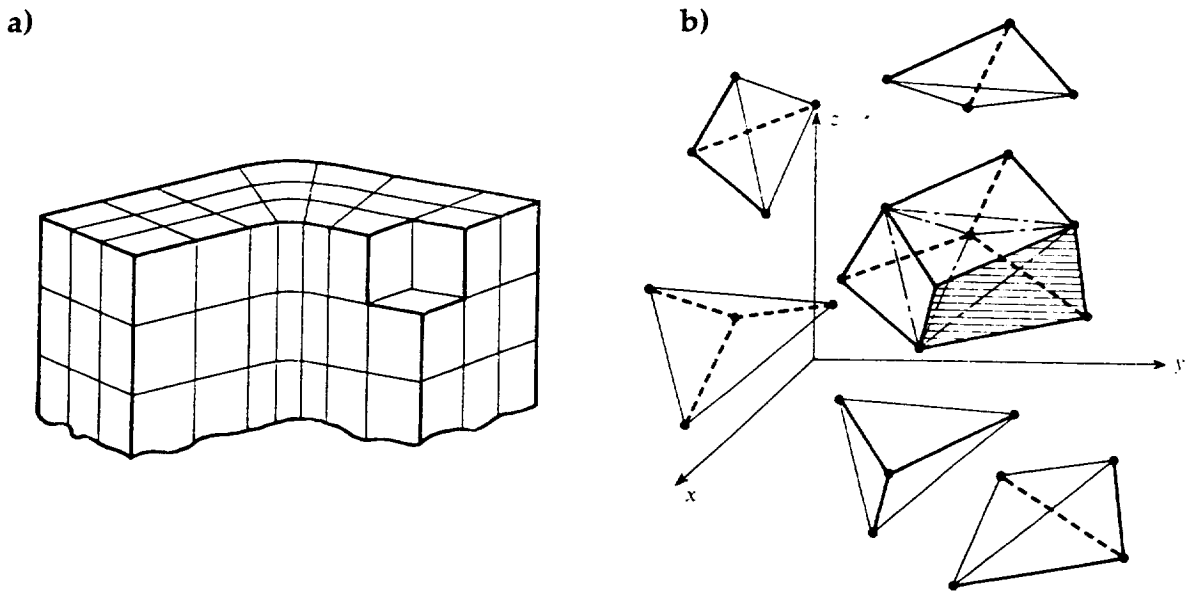


Figure 3.1 Domain partitioning into a discretization, a) hexahedra on $\Re^3$ b) composite hexahedra with eight nodes and its subdivision into five tetrahedra.

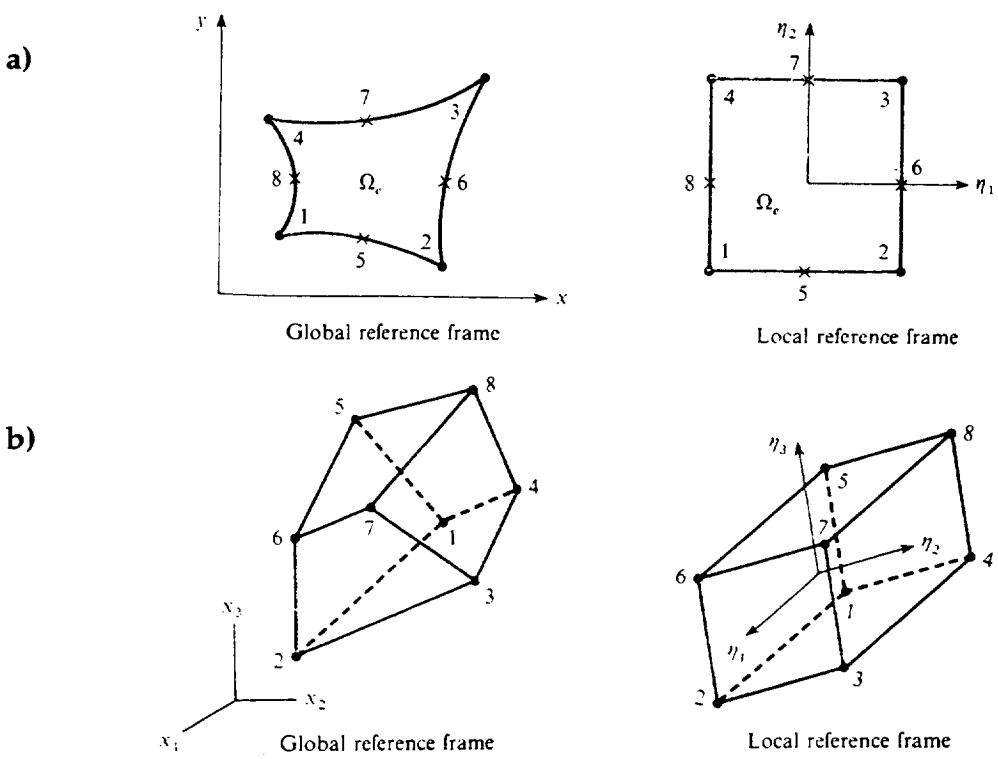


Figure 3.2 Tensor product finite element domains and node coordinate dispositions, a) two-dimensional, b) three-dimensional

permits the approximation definition (3.1) to be analytically re-expressed in continuum form as

$$q(\mathrm{x},t) \simeq q^N(\mathrm{x},t) \equiv q^h(\mathrm{x},t) = \cup q_e(\mathrm{x},t) \tag{3.10}$$

Hence, an FE approximation q^N is formed as q^h, under the finite element spatial semi-discretization Ω^h of Ω, as the union of element approximations q_e on Ω_e where $\Omega^h \equiv \cup_e \Omega_e$. On *any* (hence every) finite element domain Ω_e

$$q_e(\mathrm{x},t) \equiv \left\{N_k\left(\eta_j\right)\right\}^T \left\{Q(t)\right\}_e \tag{3.11}$$

wherein each element in the row matrix $\{N_k\}^T$ is a kth degree polynomial, and there are as many of these polynomials as there are state variable degrees of freedom on Ω_e. FE trial space basis sets $\{N_k\}$ are available for any problem dimension d, e.g., the linear bases spanning the Ω_e in Fig. 3.2 are

$$\{N_1(\eta_i)\} = \frac{1}{4}\begin{Bmatrix}(1-\eta_1)(1-\eta_2)\\(1+\eta_1)(1-\eta_2)\\(1+\eta_1)(1+\eta_2)\\(1-\eta_1)(1+\eta_2)\end{Bmatrix}_{2D} \quad , \{N_1(\eta_i)\} = \frac{1}{8}\begin{Bmatrix}(1-\eta_1)(1-\eta_2)(1-\eta_3\\(1+\eta_1)(1-\eta_2)(1-\eta_3\\(1+\eta_1)(1+\eta_2)(1-\eta_3\\(1-\eta_1)(1+\eta_2)(1-\eta_3\\(1-\eta_1)(1-\eta_2)(1+\eta_3\\(1+\eta_1)(1-\eta_2)(1+\eta_3\\(1+\eta_1)(1+\eta_2)(1+\eta_3\\(1-\eta_1)(1+\eta_2)(1+\eta_3\end{Bmatrix}_{3D} \tag{3.12}$$

3.5 Fully discrete form, algebraic statement

For any choice $\Psi_i(\mathrm{x})$, and under a spatial semi-discretization, $WS^N \Rightarrow WS^h$ is a matrix ODE system, since $\partial q^N / \partial t$ always yields $\mathrm{d}\{Q\}/\mathrm{d}t$, where $\{Q\}$ is the nodal array containing the approximation expansion coefficients $Q_j(t)$. Combining all other terms in (3.8) or (3.9) into a column array $\{RQ\}$, called "the *residual*," the weak statement integral always yields the matrix ODE system

$$WS^h = [M]\frac{\mathrm{d}\{Q\}}{\mathrm{d}t} + \{RQ\} = \{0\} \tag{3.13}$$

In (3.13), $[M]$ and $\{R\}$ denote global rank square and column matrices, respectively. and $\{Q\} \equiv \{Q(t)\}$ is the column matrix containing the *unknown* state variable

approximation coefficients at the nodes of Ω^h. The $[M]$ matrix produced by $FVWS^h$(3.8) is usually diagonal, while that produced from (3.9) via GWS^h is always non-diagonal. The *residual* $\{RQ\}$ in (3.13) is a non-linear function of $\{Q\}$, recall (2.21), since it contains contributions from *all* terms in the kinetic and dissipative flux vector approximations.

An ODE algorithm utilizes (3.13) to evaluate time derivatives created in the associated time Taylor series. For example, the θ-implicit, one-step Euler ODE algorithm over the time interval $t_{n+1} = t_n + \Delta t$ is

$$\{Q(t_{n+1})\} \equiv \{Q\}_{n+1} = \{Q\}_n + \Delta t\left(\theta \frac{d\{Q\}}{dt}\bigg|_{n+1} + (1-\theta)\frac{d\{Q\}}{dt}\bigg|_n\right) \tag{3.14}$$

Substituting WS^h (3.13) for both time derivatives in (3.14), and clearing the $[M]^{-1}$, produces the "*computable,*" fully discrete algebraic equation system

$$\{FQ\} = [M]\{Q_{n+1} - Q_n\} + \Delta t(\theta\{RQ\}_{n+1} + (1-\theta)\{RQ\}_n) = \{0\} \tag{3.15}$$

The homogeneous form (3.15) is advantageous, since the residual $\{RQ\}$ is strongly-nonlinear. The first right-side term in (3.15) is the change in $\{Q(t)\}$, over the time interval $\Delta t \equiv t_{n+1} - t_n$, which at convergence exactly matches the corresponding change in the residual $\{RQ(t)\}$ over Δt, hence $\{FQ\}$ vanishes. Attainment of this convergence requires a matrix iteration procedure, and available candidates abound cf., Varga(1967). Underlying *all* such methods is the Newton iteration algorithm

$$[JAC]\{\delta Q\}^{p+1} = -\{FQ\}^p \tag{3.16}$$

where $[JAC]$ is the *jacobian* of (3.15) and p is the iteration index. The definitions for (3.16) are

$$[JAC] \equiv \frac{\partial\{FQ\}}{\partial\{Q\}} = [M] + \theta\Delta t\frac{\partial\{RQ\}}{\partial\{Q\}}$$

$$\{Q\}_{n+1}^{p+1} = \{Q\}_{n+1}^{p} + \{\delta Q\}^{p+1} = \{Q\}_n + \sum_{i=0}^{p}\{\delta Q\}^{i+1} \tag{3.17}$$

$$\{FQ\}^p = [M]\{Q_{n+1}^p - Q_n\} + \Delta t\left(\theta\{RQ\}_{n+1}^p + (1-\theta)\{RQ\}_n\right)$$

In (3.17a), $[JAC]$ is a very large (square) non-singular matrix. As p cycles 0,1,2,..., the right-side of (3.16), i.e., (3.17c), (hopefully) becomes progressively smaller until, at some p, max $\left|\{\delta Q\}^{p+1}\right| \le \varepsilon$ for some $\varepsilon > 0$. Hence, the iteration (3.16) converges (to ε), and (3.17b) yields the fully discrete state variable nodal distribution at time t_{n+1}.

3.6 Summary

The CFD theory presented as the weak statement (3.5) *always* produces the ODE system (3.13), independent of FV/FE spatial semi-discretization specifics. This ODE statement is then employed in a time semi-discretization algorithm, hence *always* produces the strictly algebraic statement (3.15), which is non-linear unless (3.14) is selected as an *explicit* time integration algorithm, i.e., $\theta = 0$. This selection linearizes the jacobian (3.17), however $[JAC]$ remains a full square matrix via $[M]$ if an FE GWS^h semi-discretization is selected. Conversely, the FV decision yields $[M]$ diagonal, by choice, whereupon also selecting $\theta = 0$ renders the algebraic solution process (3.16) trivial. (Many explicit FE procedures for aerodynamics CFD have also artificially manipulated $[M]$ to a FV-like diagonal form, for explicit time integration efficiency.)

For $\theta \neq 0$, and either FE or FV semi-discretizations, the iteration algorithm (3.16) is fully nonlinear, and $[JAC]$ defined in (3.17) is very large! Hence, the full Newton method is rarely of use in practice, especially for 3-D solutions. The resolution is to replace (3.16)-(3.17) with a *quasi-Newton* method, which amounts essentially to replacement of (3.17a) with a smaller matrix, hence a less compute-intensive process. Methods available include stationary iterations (Picard, Gauss-Siedel, SOR), sparse matrix methods (GMRES, PCG), and block $(2k+1)$-diagonal matrix methods (AF, ADI, TP). The selected tensor product (TP) procedure for this project is detailed in a latter section.

4. WELL-POSEDNESS, STABILITY, CONVERGENCE

4.1 Overview

For Reynolds-averaged Navier-Stokes and Euler (RaNS/E) conservation law systems, the weak statement algorithm for construction of a *computable approximation* is developed. The key decisions underlying algorithm design include:

- trial space selection, hence trial space basis
- test space basis (FV or FE)
- time semi-discretization algorithm
- linear algebra iteration procedure

Once these decisions are made, the CFD algorithm/code completion process reduces to (an incredible number of) details. Prior to commission of this effort, several additional theoretical issues need resolution. For one, the RaNS/E systems are mixed initial-boundary value PDEs, for which specific well-posed boundary conditions must be determined. A Lyapunov stability analysis, Iannelli(1991), is particularly appropriate for the continuum weak statement (WS^N) construction.

Stability of a WS^N algorithm, hence its spatial semi-discretization WS^h, requires analysis for shock capturing. The "Taylor weak statement" extension to (3.5), Baker and Kim(1987), provides a continuum construction, applicable thereafter to any spatial semi-discretization. The available (linear) theory for accuracy and convergence under mesh refinement is highlighted, showing the utility and limitations of using more complete (than linear) trial space basis function sets. Thereafter, a Fourier stability analysis quantifies assessments for dissipative and dispersive error mechanisms.

This theoretical guidance supporting the cogent decision process, for construction of a specific CFD algorithm, is presented herein.

4.2. Well-posedness, boundary conditions

As commented in Section 2, the Euler form of (2.1) is thoroughly characterized for well-posed Dirichlet boundary conditions. However, this characteristics analysis is not deterministic regarding specific admissible Dirichlet constraints, which in combination with the RaNS extensions generates the analysis requirement.

Iannelli documents the well-posedness analysis for the WS^N CFD algorithm theory. The fundamental theorem (Iannelli, 1991, p. 55) proves that, a) given the Lyapunov functional $V(t,q)$ defined on $\Re^+ \times \Re^d$ with $t \in \Re^+$ and $q \in \Re^d$,and, b) for $V(t,q)$ bounded below by the norm of a continuously increasing function, and c) for

$$\left.\frac{dV}{dt}\right|_{PDE} \equiv f(t,q) - g(t,q) \tag{4.1}$$

PRECEDING PAGE BLANK NOT FILMED

where subscript "*PDE*" denotes the time derivative along a solution to $L(q)$, and, d) for f continuous and bounded above, and e), for g a positive continuous increasing function, then the solution q to $L(q)$ is bounded.

An example illustrates the utility of this theorem. The d=1 linear model of (2.1), with domain $\Re^+ \times \Omega \subset R^1$, for $x \in [a,b]$ and $t \in [t_o, \infty)$, is

$$\frac{\partial q}{\partial t} + \frac{\partial}{\partial x}(uq) = 0 \tag{4.2a}$$

For u a positive constant, the well-posed boundary condition is $q(x=a,t) = q_a(t)$, and no condition is admissible at $x=b$. A candidate Lyapunov functional for the analysis is

$$V(t,q) \equiv \frac{1}{2}\int_a^b q^2 dx \tag{4.2b}$$

Its time derivative, modulo (4.2a) is

$$\left.\frac{dV}{dt}\right|_{PDE} = \int_a^b q\frac{\partial q}{\partial t}dx = -u\int_a^b q\frac{\partial q}{\partial x}dx = \frac{u}{2}\left[q_a^2(t) - q_b^2(t)\right] \tag{4.2c}$$

For any initial condition $q(x,t=t_o)=q_o(x)$, identifying $q_a^2 \Rightarrow f$ and $q_b^2 \Rightarrow g$ in (4.1) meet the theorem requirements, hence q is bounded in R^1.

A Lyapunov functional for the RaNS/E analysis for (2.1) is developed from the "entropy functional" of Dutt(1988), defined as

$$V(q) = \rho\left[\gamma \log\left(\frac{\rho}{\bar{\rho}}\right) - \log\left(\frac{p}{\bar{p}}\right) + \frac{(\gamma-1)\bar{\rho}(m_i m_i)}{2\bar{p}\rho^2} - \gamma + \frac{p\bar{\rho}}{\bar{p}\rho}\right] + (\gamma-1)\bar{\rho} \tag{4.3}$$

The variables with superscript overbar in (4.3) denote a reference state selection such that V is a strictly positive and convex function of its argument. The Lyapunov functional then selected by Iannelli is

$$S(q,t) \equiv \int_\Omega V(q(x,t))d\tau \tag{4.4}$$

The theoretical requirement is to bound dS/dt modulo (2.1). The required differentiation, recognizing that $f=f(q)$ and using (2.1), is

$$\frac{\partial V}{\partial t} = \frac{\partial V}{\partial q}\frac{\partial q}{\partial t} = -\frac{\partial V}{\partial q}\left(\frac{\partial f_j}{\partial x_j} - \frac{\partial f_j^v}{\partial x_j}\right) \tag{4.5a}$$

Hence,

$$\frac{dS}{dt} = -\int_\Omega \frac{\partial V}{\partial q}\left(\frac{\partial f_j}{\partial x_j} - \frac{\partial f_j^v}{\partial x_j}\right)d\tau \tag{4.5b}$$

For (4.5b) and (2.2), Iannelli(1991) verifies that, for the definition

$$F_j \equiv \frac{m_j}{\rho}\left[V(q,t)-(\gamma-1)\,\bar{\rho}\right] \tag{4.6}$$

the first right-side term in (4.5b) is the divergence of a vector field F with resolution F_j as expressed in (4.6). Hence, using a Green-Gauss divergence theorem

$$\frac{dS}{dt} = -\oint_{\partial\Omega}\left(F_j - \frac{\partial V}{\partial q} f_j^v\right)\hat{n}_j d\sigma - \int_{\Omega}\frac{\partial}{\partial x_j}\left(\frac{\partial V}{\partial q}\right) f_j^v d\tau \tag{4.7}$$

where $\hat{n}_j$ is the unit normal vector on $\partial\Omega$. The domain integral (on Ω) in (4.7) can be proven positive, hence using (2.2)

$$\begin{aligned}\frac{dS}{dt} \le & -\oint_{\partial\Omega}\frac{m_j}{\rho}\left[V-(\gamma-1)\,\bar{\rho}\right]\hat{n}_j\, d\sigma \\ & +\oint_{\partial\Omega}(\gamma-1)\frac{\bar{\rho}}{\bar{p}}\frac{m_i}{\rho}\sigma_{ij}\hat{n}_j d\sigma - \oint_{\partial\Omega}(\gamma-1)\left(\frac{\bar{\rho}}{\bar{p}}-\frac{\rho}{p}\right)q_j\,\hat{n}_j\, d\sigma\end{aligned} \tag{4.8}$$

where σ_{ij} and q_j are the stress tensor and heat/flux vector contributions, respectively, in the dissipative flux vector f^v, recall (2.2) or (2.21).

Comparing (4.8) to (4.1), Iannelli(1991) identified g with the first right side term, while f is composed of the remaining two terms, which are both positive upon replacement of q_j with $-k\partial T/\partial x_j$. The closed contour integrals in (4.8) constitute evaluations on boundary segments with inflow, outflow or no through flow, selectively, as appropriate. Since surface integrals are similarly generated in the GWS^N, recall (3.9), then the proof of boundedness via (4.8) carries over directly to the weak statement formulation, hence a specific boundary condition implementation. Iannelli(1991) details the specific mix of Dirichlet boundary conditions for which (2.1), hence the weak statement algorithm (3.5), is well-posed for the Navier-Stokes definitions. Further, as Re→∞, they reduce to admissible Dirichlet constraints for the Euler form, in agreement with Table 2.1.

On inflow boundary segments, whereon $\mathbf{m}\cdot\hat{\mathbf{n}}<0$, the surface integrals in (4.8) are bounded from above and (2.1) is well-posed for the constraints

inflow (supersonic): all elements in q fixed.
inflow (subsonic): all elements in q fixed except the normal momentum component, $\mathbf{m}\cdot\hat{\mathbf{n}}$, which must be free, and

$$\begin{aligned}\sigma_{ij}\,\hat{n}_j &= a_{ij}\; u_k\,\hat{s}_{kj} - b_i \\ q_j\;\hat{n}_j &= c(T)\end{aligned} \tag{4.9}$$

where $u_k \equiv m_k/\rho$ is local velocity and $\hat{s}_{kj}$ is the resolution of the unit tangent vector lying in the surface with unit normal $\hat{\mathbf{n}}$. Hence, index "k" is correspondingly restricted

to $1 \le k \le d-1$, as is the rank of the data matrices a_{ij} and b_i. The normal heat flux may be set to a constant c, or may depend on temperature, and the typical aerodynamics situation is that all these data vanish identically.

At an outflow boundary segment, where $\mathbf{m} \cdot \hat{\mathbf{n}} > 0$, the surface integrals in (4.8) are bounded and (2.1) is well-posed for the constraints

$$\begin{aligned} &\text{outflow (supersonic): all elements in } q \text{ are free, plus} \\ &\quad \sigma_{ij}\hat{n}_j = 0 \\ &\quad q_j\hat{n}_j = 0 \end{aligned} \tag{4.10}$$

outflow (subsonic): all elements in q are free, plus

$$\begin{aligned} (p\delta_{ij} - \sigma_{ij})\hat{n}_j &= p_{out} \\ q_j\hat{n}_j &= c(T) \end{aligned} \tag{4.11}$$

Observe that as Re→∞, hence RaNS reduces to Euler equations, that (4.10)-(4.11) are consistent with the constraints in Table 2.1.

At aerodynamic surfaces, the Euler formulation requirement is flow tangency, hence $\mathbf{m} \cdot \hat{\mathbf{n}} = 0$ This is augmented with no-slip ($\mathbf{m} \cdot \hat{\mathbf{s}} = 0$) in the Navier-Stokes formulation. Therefore, on no-through flow boundary segments, the surface integrals in (4.8) are bounded, hence (2.1) is well-posed, for the conditions:

$$\begin{aligned} \text{wall (inviscid):} \quad &(\hat{\mathbf{n}} \cdot \nabla)q = 0 \text{ for all elements in } q, \text{ plus} \\ &\mathbf{m} \cdot \hat{\mathbf{n}} = 0 \end{aligned} \tag{4.12}$$

$$\begin{aligned} \text{wall (viscous):} \quad &(\mathbf{n} \cdot \nabla)q = 0 \text{ for elements } \rho \text{ and } E, \text{ and} \\ &m_j = 0 \text{ for } 1 \le j \le d \\ &q_j\hat{n}_j = c(T) \end{aligned} \tag{4.13}$$

In summary, (4.9)-(4.13) are admissible Dirichlet data specifications on bounding surface distributions of momentum flux, deviatoric traction and heat flux vector, for the RaNS/E conservation law system (2.1). These data specifications collectively prevent the growth of all integrals in (4.8), except the first right-side term involving convective fluxes, i.e.,

$$\oint_{\partial\Omega} \frac{m_j}{\rho}\left[V - (\gamma - 1)\,\bar{\rho}\right]\hat{n}_j \, d\sigma \tag{4.14}$$

which is allowed to vary according to the constraints presented in Table 2.1. Consequently, the hypothesis of the Lyapunov stability theorem is satisfied, hence the solution q is bounded. Importantly, the Galerkin weak statement GWS^N theory (3.9)

explicitly exposes, in the continuum, these surface integrals for the resultant CFD algorithm construction.

4.3 Stability, artificial dissipation

Solutions $q(\mathbf{x},t)$ to the RaNS/E form of (2.1) exist that can exhibit singular character, e.g., shocks, contact discontinuities and rarefaction waves. Further, CFD simulations of RaNS/E aerodynamics statements generate only a discretized approximation q^h to q, and dispersive error mechanisms exist to destabilize the solution as well as the linear algebra process. Therefore, an augmentation to the CFD theoretical process is required to identify *artificial* dissipative processes, typically termed "*numerical diffusion,*" to stabilize a q^h-generation procedure.

Meeting this requirement with precision has occupied the aerodynamics CFD theoretical community for decades, leading to creation of a multitude of "dissipative CFD algorithms" ranging from explicit artificial viscosity, cf, Pulliam (1986), Jameson(1982), to flux-difference, flux-vector and flux eigen-vector splittings with upwind differencing, cf., Steger and Warming(1981), Liou and vanLeer (1988), Roe(1981), Lombard, *et al* (1982), etc. A cogent review of these methods with distinguishing characterization is published by Vinokur(1990).

The requirement exists therefore to generalize the weak statement theory for artificial dissipation. In distinction to the historical procedures, which all require a spatial semi-discretization to support theoretical musings, a development in the continuum would be universally applicable for arbitrary trial space basis choice. The progenitor of the "*Taylor weak statement* (*TWS*)" theory is of Lax-Wendorff origin, as first developed for FE methods by Donea (1984) for convective problems. The generalization for the RaNS/E model problem class, as developed by Baker and Kim (1987), yielded a theory encompassing *sixteen* independently derived dissipative CFD algorithms as special cases. For the RaNS/E problem statement, Iannelli (1991) and Freels (1992), independently developed and verified specific *TWS* dissipation mechanisms.

The *TWS* formulation recognizes that dissipation in the absence of physical diffusion is the requirement. Hence, $\mathbf{f}^v$ and s in (2.1) vanish for the theoretical development, but return at completion. Recalling that $\mathbf{f}=\mathbf{f}(q)$, the analysis departure point is the Euler form of (2.1), *i.e.*,

$$L^E(q) = \frac{\partial q}{\partial t} + \frac{\partial f_j}{\partial x_j} = \frac{\partial q}{\partial t} + \frac{\partial f_j}{\partial q}\frac{\partial q}{\partial x_j} = \frac{\partial q}{\partial t} + A_j \frac{\partial q}{\partial x_j} = 0 \qquad (4.15)$$

For development simplicity, the coordinate transformation (2.23) is suppressed, hence A_j defined in (4.15) is the cartesian Euler flux vector jacobian comparable to (2.28), with

its eigenvalue composition. Following Lax-Wendroff, a temporal semi-discretization yields

$$q(t_{n+1}) \equiv q^{n+1} = q^n + \left(\frac{\partial q}{\partial t}\right)^n \Delta t + \left(\frac{\partial^2 q}{\partial t^2}\right)^n \frac{\Delta t^2}{2} + \left(\frac{\partial^3 q}{\partial t^3}\right)^n \frac{\Delta t^3}{6} + O(\Delta t^4) \tag{4.16}$$

on the interval $\Delta t \equiv t_{n+1} - t_n$. Then, (4.15) enables derivative interchanges yielding (at time t_n)

$$\frac{\partial q}{\partial t} = -\frac{\partial f_i}{\partial x_i} \tag{4.17}$$

$$\frac{\partial^2 q}{\partial t^2} = \frac{\partial}{\partial t}\left(\frac{\partial q}{\partial t}\right) = -\frac{\partial}{\partial t}\left(\frac{\partial f_i}{\partial x_i}\right) = -\frac{\partial}{\partial x_i}\left(\frac{\partial f_i}{\partial t}\right) = -\frac{\partial}{\partial x_i}\left(\frac{\partial f_i}{\partial q}\frac{\partial q}{\partial t}\right) \tag{4.18}$$

$$= -\frac{\partial}{\partial x_i}\left(A_i \frac{\partial q}{\partial t}\right) = \frac{\partial}{\partial x_i}\left(A_i A_j \frac{\partial q}{\partial x_j}\right) \equiv \frac{\partial}{\partial x_i}\left(A_i\left(\alpha\frac{\partial q}{\partial t} + \beta A_j \frac{\partial q}{\partial x_j}\right)\right)$$

$$\frac{\partial^3 q}{\partial t^3} = \cdots \equiv \frac{\partial}{\partial x_i}\left(A_i \frac{\partial}{\partial x_j} A_j\left(\gamma\frac{\partial q}{\partial t} + \mu A_k \frac{\partial q}{\partial x_k}\right)\right) \tag{4.19}$$

The terminal forms in (4.18) and (4.19) indicate the arbitrariness for convex combinations $-\alpha + \beta = 1$ and $-\gamma + \mu = 1$.

Collecting terms by like coefficient, substituting (4.17)-(4.19) into (4.16), and taking the limit $\Delta t \to \varepsilon \geq 0$ yields reexpression of (2.1) as

$$\begin{aligned} L^m(q) \equiv & \left[1 - A_i \frac{\partial}{\partial x_i}\left(\alpha\frac{\Delta t}{2} + \gamma\frac{\Delta t^2}{6} A_j \frac{\partial}{\partial x_j}\right)\right]\frac{\partial q}{\partial t} - s \\ & + \frac{\partial}{\partial x_i}\left[f_i - f_i^v - A_i\left(\beta\frac{\Delta t}{2} A_j \frac{\partial q}{\partial x_j} + \mu\frac{\Delta t^2}{6} A_j \frac{\partial}{\partial x_j} A_k \frac{\partial q}{\partial x_k}\right)\right] = 0 \end{aligned} \tag{4.20}$$

where superscript m denotes "Taylor-series modified." The dominant dissipation term in (4.20) has the β premultiplier, hence the simplest dissipative form for (4.20) is

$$L^m(q) = L(q) - \frac{\Delta t}{2}\frac{\partial}{\partial x_j}\left(\beta A_j A_k \frac{\partial q}{\partial x_k}\right) = 0 \tag{4.21}$$

Equation (4.21) clearly indicates that via the temporal semi-discrete process, a second-order (parabolic) perturbation becomes appended to the RaNS/E system, that involves an outer (tensor) product of flux vector jacobian matrices with the gradient of the state variable. For any finite Δt, this term is scaled in magnitude by the coefficient

β. Since this augmentation is in the continuum, then the entire content of Chapter 3 regarding weak statement algorithm development is unmodified, except for replacement of $L(q^N)$ in (3.5) with $L^m(q^N)$, (4.20) or (4.21). Thereby, the ODE system (3.13) remains the end point of any weak statement spatial semi-discretization, while (3.15)-(3.18) remain the matrix algebraic terminal statement for any time discretization (3.14). Thereby, the theoretical crutch of (4.16) being explicit, and completed for $L^E(q)$, does not yield a practical constraint.

The "β-term" in (4.21) represents a significant complication since the matrix product $A_j A_k$ couples all PDEs in the modified RaNS/E system. Therefore, additional simplifying assumptions may be appropriate for computational tractability. With the generalized coordinate transformation in (4.21), the tensor product produces an order (d+2) matrix containing products of all terms in (2.28),c.f., Iannelli(1991).

$$A_j A_k \simeq u_j u_k = e_{ji} e_{k\ell} v_i v_\ell \tag{4.22}$$

where $v_k \equiv m_k / \rho$ (and v_ℓ) is the contravariant velocity resolution, recall (2.25).

Substitution of (4.22) into (4.21 yields a "tensor diffusivity" form that has been reported in the atmospheric sciences literature, c.f., Gresho(1989). The explicit appearance of Δt in (4.21) can be removed (approximately) via definition of scalar Courant number $C_{\underline{\gamma}} \equiv u_{\underline{\gamma}} \Delta t / h_{\underline{\gamma}}$, where subscript $\underline{\gamma}$ is not for summation. Then lumping $\beta C_{\underline{\gamma}} / 2$ into a new (distributed) coefficient $\beta(\underline{x})$, another form for (4.21)-(4.22) is

$$L^m(q) = L(q) - \beta(x) h \frac{\partial}{\partial x_j}\left(u_j \hat{u}_k \frac{\partial q}{\partial x_k} \right) = 0 \tag{4.23}$$

where h is a length scale (eventually a mesh measure) and $\hat{u}_k$ is the velocity unit vector.

In (4.23), the tensor diffusivity $u_j \hat{u}_k$ may contain negative products dependent on the scalar resolution (coordinate system). Strict positivity can be assured by contracting over j and k and taking magnitudes, yielding the replacement

$$\mathbf{v} \equiv \left| \frac{e_{jk} e_{j\ell} v_k v_\ell}{\mathbf{u}} \right| \tag{4.24}$$

where $\mathbf{u}$ is the velocity magnitude. Replacement of the contracted tensor product in (4.23) by $\mathbf{v}$ can rigidly enforce *positivity* if required for (4.21) to be stable. Thereby, the parabolic perturbation (β) term in (4.23) is guaranteed uniformly dissipative, and in either case the scale level is controlled by $\beta \geq 0$. The Fourier stability analysis quantifying this construction is detailed later in this section.

4.4. Accuracy, asymptotic convergence

The augmentation of (2.1)-(2.22) to the form (4.23) defines the RaNS/E system for which the weak statement approximate solution construction is to be completed. To proceed further requires a firm decision to on FV vs FE semi-discrete implementation, of (3.5), and the latter is selected. Thereby, the fundamental choice resides in trial space basis, both with respect to element domain shape (triangle vs quadrilateral, tetrahedron vs hexahedron) and the degree k of the embedded polynomials. As discussed, the selection is quad and hex, spanned by the FE tensor product basis $\{N_k(\eta)\}$.

Guidance for choosing other than the simplest form (k=1, recall (3.12), is provided by available asymptotic convergence estimates. For the linearized parabolic model problem, c.f., Oden and Reddy(1976, Ch.9), upon resolution of error into spatial and time (truncation) error contributions, and using the triangle inequality, the semi-discrete contribution to (3.2), *i.e.*,

$$e^h(\mathbf{x},t) \equiv q(\mathbf{x},t) - q^h(\mathbf{x},t) \tag{4.25}$$

is bounded under FE mesh refinement, at any time t_n, in the form

$$\|e^h(t_n)\|^2_{H^1(\Omega)} \le C_1 h_e^{2k} \|q(t_n)\|^2_{H^{k+1}(\Omega)} + C_2 \Delta t^{f(\theta)} \|q(t_o)\|^2_{H^1(\Omega)} \tag{4.26}$$

where C_1 and C_2 are constants (for sufficiently refined mesh). The *key* issue expressed in (4.26) is that the spatial semi-discretization error contribution is bounded by the extremum *measure* of the mesh h_e raised to twice the power k, the completeness degree of $\{N_k\}$. Specifically, under (uniform) mesh refinement, and as measured in the H^1 Sobolev norm squared, (4.26) predicts that the linear basis algorithm is analogously "second-order accurate," and it becomes asymptotic "higher-order accurate" for k>1.

The norms involved in estimating asympotic convergence, e.g., (4.26), are H^r Sobolev norms with basic definition

$$\|u\|^2_{H^r(\Omega)} \equiv \sum_{i=0}^{r} \sum_{j=0}^{r} \int_\Omega \frac{\partial^i u}{\partial x_k^i} \frac{\partial^j u}{\partial x_k^j} d\tau \tag{4.27}$$

Equation (4.27) states that the function $u(\mathbf{x})$ is sufficiently well-behaved (*smooth*), such that all products of derivatives on the range $0 \le (i,j) \le r$ are square-integrable, i.e., they exist. Hence, the semi-discrete error e^h estimate (4.26) lies in H^1 on Ω, i.e., all products of first derivatives are "smooth enough" to be integrated. Various norms of the exact solution exist in the right side of (4.26) to modulate the actual "size" of the error. Specifically, the first right-side norm states that the analytical solution $q(\mathbf{x},t=t_n)$ must possess k+1 derivatives, which for basis selection k>1 requires greater smoothness than

does the RaNS/Euler PDE system itself! Secondly, the initial condition $q(x,t_0)$ must have square-integrable first derivatives.

The predictive appropriateness for (4.26) for parabolic forms of laminar and turbulent RaNS systems is verified, Baker(1983, Ch. 6). However, in aerodynamics, the subject RaNS/E systems can possess non-smooth solutions, hence CFD approximate methods will use artificial dissipation mechanisms for stability. Thereby, in this situation, physical dissipation via f^v plays essentially no stability role, hence Re is not a factor.

The companion linearized theoretical analysis for smooth solutions to the Euler model problem predicts the asymptotic error estimate form as

$$\begin{aligned}\left\|e^h(t_n)\right\|^2_{H^1} \le C_1 h_e^{2k}\left\|q(t_n)\right\|^2_{H^{k+1}(\Omega)} + C_2\Delta t^2\left\|q(t_0)\right\|^2_{H^1(\Omega)} \\ + C_3 h_e^2\int_{t_0}^{t_n}\left\|q(t)\right\|^2_{H^{k+1}(\Omega)}dt\end{aligned} \tag{4.28}$$

Thereby, via the C_3-term, in (4.28), the asymptotic convergence rate for smooth solutions for Euler, or RaNS in the limit Re$\rightarrow\infty$, is *independent* of the FE basis degree k.

Select model problem numerical convergence studies indeed verify (4.28), Baker(1983, Ch.4). Hence, use of $k>1$ basis appears an option that may be of limited utility in inviscid aerodynamics. Therefore, viewing (4.26) and (4.28), the project decision was made to restrict TWS^h implementation to the linear tensor product FE basis on quads and hexahedra only.

4.5 Stability, artificial dissipation

The requirement is to quantize stability, for the TWS^h modified conservation law form (4.20). The restriction to d=1 (and k=1) directly facilitates a discrete Fourier stability analysis for the model problem.

$$L(q) = \frac{\partial q}{\partial t} + \frac{\partial}{\partial x}\left(uq - \varepsilon\frac{\partial q}{\partial x}\right) = 0 \tag{4.29}$$

where u is speed and ε is identified with Re^{-1}, recall (2.21).

The Fourier representation of the analytical solution $q(x,t)$ to (4.29) is

$$q(x,t) = \sum_k^{\infty} B_k e^{i(\omega_k x - \alpha_k t)} \tag{4.30}$$

In (4.30), ω_k is the wave number of the kth mode, with wavelength $\lambda_k = 2\pi/\omega_k$. B_k is the expansion coefficient set, and $i=\sqrt{-1}$ is the imaginary unit. Since (4.29) is linear, substituting (4.30) for the case $\varepsilon = 0$ yields $\alpha_k = -u\omega_k$. Since k is thereby a free index in (4.30), all Fourier modes for $q(x,t)$ are of the form

$$q(\omega,x,t) \equiv q_\omega(x,t) = e^{i\omega(x-ut)} \tag{4.31}$$

The analysis assumption is that q^h determined via a TWS^h algorithm possesses the similar Fourier representation. Hence, x becomes discretely represented as $j\Delta x$, and the generic form is

$$q^h_\omega(j\Delta x,t) \equiv e^{i\omega(j\Delta x - vt)} \tag{4.32}$$

where $v = v_R + iv_I$ is the (complex) modal speed of the Fourier mode, and both v_R and v_I are real functions. Multiplying (4.32) through by $e^{-i\omega ut}$ yields

$$q^h_\omega(j\Delta x,t) = e^{\omega t(v_I + i(u - v_R))} q_\omega(j\Delta x,t) \tag{4.33}$$

hence q^h_ω and q_ω differ by the indicated complex function.

Since by definition $q(x,t) = q^h(x,t) + e^h(x,t)$, then using (4.33), the TWS^h semi-discrete approximation error Fourier representation is

$$e^h_\omega = q_\omega - q^h_\omega = (1 - e^{\omega t(v_I + i(u - v_R))}) q_\omega \tag{4.34}$$

which provides insight into the consequence of a spatial semi-discretization. Specifically, if the phase speed v is identical to u, then $v_R = u$ and $v_I = 0$, hence $e^h_\omega = 0$ for all ω and t. However, for a practical q^h, typically $v_R < u$, especially at large wave number (short wave-length), while v_I can be made nominally zero, prior to specific addition of artificial diffusion, whereupon $v_I < 0$.

The Fourier analysis platform is complimental by combining the TWS^h statement (4.33) with a time Taylor series, recall (4.14). Over the n-step elapsed time interval $n\Delta t = t_n - t$, with a capital letter denoting fully discrete

$$\begin{aligned} Q^n_j \equiv Q(j\Delta x, t + n\Delta t) &= e^{i\omega(j\Delta x - v(t + n\Delta t))} \\ &= e^{i\omega v(n\Delta t)} e^{i\omega(j\Delta x - vt)} \\ &\equiv g^n q^h_\omega(j\Delta x,t) \end{aligned} \tag{4.35}$$

Thus, the complex *"amplification factor,"* $g = g_R + ig_I$, completes the semi-discrete transition to the fully discrete approximation Fourier representation.

The Fourier analysis process thereby seeks solutions for (the complex functions) phase speed v, and amplification factor g, dependent upon the choices exercised by the CFD algorithm designer for (4.29), and or any extensions/modifications thereto. For the Euler/Navier-Stokes model (4.29), the jacobian of the kinetic flux vector $f = uq$ is u, hence for (4.20) and (4.29)

$$L^m(q) = L(q) - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(\alpha u \frac{\partial q}{\partial t} + \beta u^2 \frac{\partial q}{\partial x}\right) - \frac{\Delta t^2}{6}\frac{\partial}{\partial x}\left(u\frac{\partial}{\partial x}\left(\gamma u \frac{\partial q}{\partial t} + \mu u^2 \frac{\partial}{\partial x}\left(u \frac{\partial q}{\partial x}\right)\right)\right) \tag{4.36}$$

Substituting (4.36) into TWS^N (3.9) then yields

$$\begin{aligned} TWS^N &= \int_\Omega \psi_i(x) L^m\left(q^N\right) dx \\ &= \int_\Omega \psi_i \left(\frac{\partial q^N}{\partial t}\right) dx + \int_\Omega \frac{d\psi_i}{dx}\left[-uq^N + \varepsilon \frac{\partial q^N}{\partial x}\right] dx \\ &+ \frac{\Delta t^2}{2}\int_\Omega \frac{d\psi_i}{dx}\left[\alpha u \frac{\partial q^N}{\partial t} + \beta u^2 \frac{\partial q^N}{\partial x}\right] dx \\ &+ \frac{\Delta t^2}{6}\int_\Omega \frac{d\psi_i}{dx}\left[\gamma u \frac{\partial}{\partial x}\left(u \frac{\partial q^N}{\partial t}\right) + \mu u^2 \frac{\partial}{\partial x}\left(u \frac{\partial q^N}{\partial x}\right)\right] dx \\ &+ \psi_i \left[uq^N - \varepsilon \frac{\partial q^N}{\partial x} + \frac{\Delta t}{2}(\cdot) + \frac{\Delta t^2}{6}(\cdot)\right]_{x_L}^{x_R} \equiv 0 \end{aligned} \tag{4.37}$$

The goal is to determine the Fourier characterization of (4.37). Hence, the last term, representing the surface integrals created via the Green-Gauss theorem, is discarded (since interest lies in stability at a representative interior node). Further, the "gamma" term in the fourth integral is discarded, since the FE basis equivalents of Ψ_i in H^1 are usually not twice differentiable. Then, limiting illustration to the linear (k=1) basis, the FE spatial semi-discretization form is

$$\begin{aligned} q(x,t) &\simeq q^N(x,t) \equiv \bigcup_e q_e(x,t) \\ q_e(x,t) &\equiv \{N_1\}^T \{Q(t)\}_e = \frac{1}{2}\{1-\eta, 1+\eta\}\{Q\}_e \end{aligned} \tag{4.38}$$

The resulting semi-discrete form of TWS^N, i.e., TWS^h, is

$$TWS^h = \int_\Omega \psi_i(x) L^m(q^h)\mathrm{d}x = \underset{e}{S}\left(\int_{\Omega_e} \{N_1\} L^m(q_e)\mathrm{d}x\right)$$

$$= \underset{e}{S}\left([M(\alpha,\gamma,\Delta t)]_e \frac{\mathrm{d}\{Q\}_e}{\mathrm{d}t} + \{RQ(\beta,\Delta t,u,\varepsilon)\}_e\right)$$

$$= [M]\frac{\mathrm{d}\{Q\}}{\mathrm{d}t} + \{RQ\} = \{0\} \tag{4.39}$$

which locates the TWS^h parameter set (α,β,γ) within the generic expression (3.13).

The basic building of the Fourier stability analysis blocks are the "*element matrices*" created for every term in (4.39). Proceeding in the order of (4.37), and substituting (4.38) as needed, the first term in (4.39) is

$$h_e[M]_e\{Q\}_e' \equiv \int_{\Omega_e} \{N_1\}\frac{\partial q_e}{\partial t}\mathrm{d}x = \int_{\Omega_e} \{N_1\}\{N_1\}^T \mathrm{d}x\{Q\}_e' = \frac{h_e}{6}\begin{bmatrix} 2,1 \\ 1,2 \end{bmatrix}\{Q\}_e' \tag{4.40}$$

where superscript prime denotes ordinary derivative and h_e is element length. Next,

$$U_e[V]_e\{Q\}_e \equiv \int_{\Omega_e} \frac{d\{N_1\}}{dx} u_e q_e dx = \frac{U_e}{2}\begin{bmatrix} -1,-1 \\ 1,\ 1 \end{bmatrix}\{Q\}_e \tag{4.41}$$

where U_e is the average velocity in element Ω_e. Similarly

$$\frac{\varepsilon}{h_e}[D]_e\{Q\}_e \equiv \int_{\Omega_e} \varepsilon \frac{d\{N_1\}}{dx}\frac{\partial q_e}{\partial x} dx = \frac{\varepsilon}{h_e}\begin{bmatrix} 1,-1 \\ -1,\ 1 \end{bmatrix}\{Q\}_e \tag{4.42}$$

is the diffusion contribution with $\varepsilon \geq 0$. Hence, (4.40)-(4.42) identify the basic algorithm element matrix library.

The terms remaining in (4.37) involve other multipliers on the mass matrix $[M]_e$, the velocity matrix $[V]_e$, and the diffusion matrix $[D]_e$. Dividing through by the element measure h_e, and for element Courant number $C_e \equiv U_e\Delta t/h_e$, (4.40) written on the *generic* FE domain Ω_e becomes

$$\begin{aligned} TWS_e = &\left([M]_e + \frac{\alpha C_e}{2}[V]_e + \frac{\gamma C_e^2}{6}[D]_e\right)\{Q\}_e' \\ &+ \frac{U_e}{h_e}\left(-[V]_e + \frac{\varepsilon}{h_e U_e}[D]_e + \frac{\beta C_e}{2}[D]_e\right)\{Q\}_e \end{aligned} \tag{4.43}$$

The final step is to use (4.43) as the time-derivative expression in the temporal Taylor series (3.14). The resultant algebraic equivalent of (3.15) on the generic FE domain Ω_e is, upon substituting (4.43)

$$
\begin{aligned}
\{FQ\}_e = &\left([M]_e + \frac{\alpha C_e}{2}[V]_e + \frac{\gamma C_e^2}{6}[D]_e\right)\{Q_{n+1} - Q_n\}_e \\
&+ C_e\theta\left(-[V]_e + \frac{\varepsilon}{h_e U_e}[D]_e + \frac{\beta C_e}{2}[D]_e\right)\{Q_{n+1}\}_e \\
&+ C_e(1-\theta)\left(-[V]_e + \frac{\varepsilon}{h_e U_e}[D]_e + \frac{\beta C_e}{2}[D]_e\right)\{Q_n\}_e
\end{aligned}
\tag{4.44}
$$

which is the generic element-rank contribution to (3.15), the TWS^h global order matrix statement. Note that the "Δt" multiplier in (3.15) is embedded in the element Courant number (C_e) residual multiplier in (4.44).

The final step to Fourier characterization is completing the linear algebra statement (3.17) for (4.44) representing (3.15). Since the analysis is linearized, only one iteration occurs in (3.16), hence $\{\delta Q\}^{p+1} \Rightarrow \{\Delta Q\} = \{Q_{n+1} - Q_n\}$ for p=0. The jacobian is easy to form from its definition (3.17a), using (4.44), hence the computational matrix statement for (4.44) is

$$
\begin{aligned}
&S_e\left([JAC]_e\{\Delta Q\}_e = -\Delta t\{R\}_n\right) \\
&\qquad [JAC]_e = [M]_e + C_e(\theta - \alpha/2)[V]_e + \frac{C_e\theta\varepsilon}{h_e U_e}[D]_e + C_e^2\left(\frac{\theta\beta}{2} + \frac{\gamma}{6}\right)[D]_e \\
&\qquad \{RQ\}_n = C_e\left(-[V]_e + \frac{\varepsilon}{h_e U_e}[D]_e + \frac{\beta C_e}{2}[D]_e\right)\{Q\}_e^n
\end{aligned}
\tag{4.45}
$$

Viewing (4.45) confirms the role that the TWS^h parameter set (α,β,γ) plays in the algorithm. The α and γ terms reside only in the linear algebra jacobian, premultiplied by Courant number C_e, as modifications to the convection and diffusion terms created by $L(q)$. In distinction, the β term resides in both sides of (4.45) as augmentation to the natural diffusion term (with ε).

The Fourier modal solutions for v and g are generated via replacement of the global assembly form (4.45) with assembly over the FE element pair sharing node "j" in the d=1 *mesh, i.e.,* $x_j = x_L + j\Delta x$, and for h_e and U_e assumed constants. For example, from the linear FE basis definition of $[M]_e$ in (4.40)

$$
\begin{aligned}
\underset{e}{S}\left(M\{Q\}_e\right)_j &\Rightarrow \frac{1}{6}\begin{bmatrix} 2 & 1 & \\ 1 & 2 & 1 \\ & 1 & 2 \end{bmatrix}\begin{Bmatrix} Q_{j-1} \\ Q_j \\ Q_{j+1} \end{Bmatrix} = \frac{1}{6}(\,Q_{j-1} + 4Q_j + Q_{j+1}\,) \pm \frac{1}{6}\left(2Q_j\right) \\
&= Q_j + \frac{1}{6}\left(Q_{j-1} - 2Q_j + Q_{j+1}\right) = Q_j + \frac{1}{6}\delta^2 Q_j
\end{aligned}
\tag{4.46}
$$

where δ^2 is the second-order accurate FD diffusion operator. The assembly of $[D]_e\{Q\}_e$ also produces δ^2 whereas the second-order accurate FD operator $\Delta = \frac{1}{2}(\cdot)_{j+1} - (\cdot)_{j-1}$ is produced by assembly of $-[V]_e\{Q\}_e$.

Hence, the FD operator-equivalent form of (4.45), for the k=1 basis, uniform mesh and constant convection U_e, is

$$\left[1+\alpha_A\Delta+\left(\frac{1}{6}+\alpha_B\right)\delta^2\right]\Delta Q_j=-C\left(\Delta+\alpha_D\delta^2\right)Q_j^n \tag{4.46}$$

with the definitions

$$\begin{aligned}\alpha_A &\equiv C(\theta-\alpha/2)\\ \alpha_B &\equiv C\theta\left(\frac{\varepsilon}{hU}+C\left(\frac{\theta\beta}{2}+\frac{\gamma}{6}\right)\right)\\ \alpha_D &\equiv C\left(\frac{\varepsilon}{hU}+\frac{\beta C}{2}\right)\end{aligned} \tag{4.47}$$

Substituting the Fourier modal form (4.32), and (4.35) for n=1, yields the explicit expression for the amplification factor g for the linear basis TWS^h algorithm as

$$g=1-\frac{C\left(\Delta+\alpha_D\delta^2\right)}{\left(1+\alpha_A\Delta+\left(\frac{1}{6}+\alpha_B\right)\delta^2\right)} \tag{4.48}$$

Baker and Kim(1987) report the solution for (4.48) as the resolution of g into real and imaginary components in the form

$$g_R=\frac{C}{\omega\Delta t}\left[m^2\left(\frac{C}{2}-\alpha_1\right)+m^4\left(-\frac{C^3}{8}+\frac{\alpha_1C^2}{2}-\left(\beta_2-\beta_0\right)C+\left(\beta_1-\beta_0\alpha_1\right)\right)+O\left(m^6\right)\right] \tag{4.49}$$

$$\begin{aligned}g_I &= U-\frac{C}{\omega\Delta t}\geq\left[m^3\left(\frac{C^2}{3}-\alpha_1C+(\beta_1-\beta_0)\right.\right.\\ &\left.\left.+m^5\left(-\frac{C^4}{30}+\frac{\alpha_1C^3}{6}-\frac{(\beta_2-\beta_0)}{2}C^2+(\beta_1-\beta_0\alpha_1)C-\beta_0(\beta_2-\beta_0)\right)+O\left(m^7\right)\right]\end{aligned} \tag{4.50}$$

where mesh parameter definition is $m=\omega\Delta x=2\pi\Delta x/\lambda$, where λ is the mode wavelength, and C is the (uniform) Courant number. The subscripted Greek parameters α_j and β_j in (4.49)-(4.50) are detailed algebraic functions of the parent TWS^h parameter set (α,β,γ).

Algorithm dissipation, hence stability (augmentation), is dominated by the coefficient of the lowest power (m^2) term in (4.49). Substituting the definition $\alpha_1=\alpha_A+\alpha_D$ yields

$$\frac{C}{2}-\alpha_1=C\left[\frac{1}{2}-\theta+\frac{\alpha}{2}-\frac{\beta C}{2}-\frac{\varepsilon}{hU}\right] \tag{4.51}$$

Hence, solution process dissipation is the balance between the *physical diffusion*, parameterized by ε/hU, and the combination of θ, α and β. For θ=1/2, and α=βC, numerical dissipation is absent at this level. The parameter ε/hU is proportional to

inverse Reynolds number, which is small unless hU, the local mesh measure times speed, is the same order as inverse Re, an unlikely occurrence. Hence, both $\theta > 1/2$ and $\beta > 0$ in (4.51), will dominate algorithm stability.

Baker and Kim(1987) report more than a dozen dissipative CFD algorithm constructions that match the TWS^h formulation (4.37) for select specific $(\alpha,\beta,\gamma,\mu)$, including Euler algorithms composed of upwind flux vector splittings. Table 4.1 summarizes the comparative results, by terms in the TWS^h operator equation (4.46).

Current research (Chaffin and Baker, 1994) has succeeded in expanding the scope of this Fourier analysis to include quadratic and cubic FE bases. A manipulation with the amplification factor solution produces the phase velocity ϕ defined as

$$\phi \equiv \frac{C}{\omega \Delta x} \tan^{-1}\left[g_I / g_R\right] \tag{4.52}$$

from which one can determine the relative error in propagation speed of the Fourier mode of wave number ω. Figure 4.1 summarizes the computed error in amplification factor $(1-g)$ and the error in phase velocity $(U-\phi)$ for the range of FD, FV and FE WS^h and TWS^h $(\beta>0)$ algorithms. The FE constructions are certainly competitive with the reference algorithms, and benefits to use of higher degree FE basis functions are firmly quantized in wave number space.

4.6 Summary

This section has examined the critical issues of algorithm well-posedness, convergence and stability. The NS form is proven a parabolic perturbation on the Euler hyperbolic conservation law form. Via definition of a suitable Lyapunov function, and using the weak statement construction, a set of well-posed boundary conditions are established for the NS system that default to admissible Euler conditions. The generic WS construction was then augmented, via a Lax-Wendroff time semi-discretization, to expand the CFD algorithm statement for a wide class of dissipative constructions. Following a summary of available asymptotic error estimates under discretization refinement, the TWS^h construction for d=1 and the linear basis (k=1) was completed for Fourier modal assessment of phase error and artificial dissipation. A summary of the theoretical extension to k=2,3 FE bases completes the section.

Table 4.1 Summary of CFD algorithms within Taylor weak statement, Baker & Kim (1987)

Method	Reference	θ	α_A	$(\frac{1}{6}-\alpha_B)$	α_C	$-\alpha_D$	α_E	Comments
Taylor weak statement	This paper, equation (32)	all	$c\left(\theta-\frac{\bar{\alpha}}{2}-\frac{\bar{\gamma}}{3}\Delta t a_x\right)$	$\frac{1}{6}-c^2\left[\frac{\bar{\gamma}}{6}+\theta\left(\frac{\bar{\beta}}{2}+\frac{\bar{\mu}}{2}\Delta t a_x\right)\right]$	$c^3\theta\frac{\bar{\mu}}{6}$	$-c\frac{\bar{\beta}}{2}+\frac{\bar{\mu}}{2}\Delta t a_x$	$c^2\frac{\bar{\mu}}{6}$	$\bar{\alpha},\bar{\beta},\bar{\gamma},\bar{\mu},\theta$ to be defined
Bubnov Galerkin	This paper, equation (54)	all	$c\theta$	$\frac{1}{6}$	0	0	0	$\bar{\alpha}=0=\bar{\beta}=\bar{\mu}=\bar{\gamma}$
Donor-cell upwind	12	0	0	0	0	$-\frac{(\operatorname{sgn} a)}{2}$	0	$\lvert c\rvert\leqslant 1$
Lax-Wendroff	13	0	0	0	0	$-\frac{c}{2}$	0	$\lvert c\rvert\leqslant 1$
Euler-Taylor-Galerkin	9	0	$c\bar{\theta}$	$\frac{1}{6}-\frac{c^2}{6}(1-3\bar{\theta})$	0	$-c(\frac{1}{2}-\bar{\theta})$	0	$\bar{\theta}=0$ for ETG, $\bar{\theta}=\frac{1}{2}$ for CN-TG
Euler-characteristic-Galerkin	8	0	0	$\frac{1}{6}$	0	$-\frac{c}{2}$	$\frac{c^2}{6}$	$\lvert c\rvert\leqslant 1$ for this form
Swansea-Taylor-Galerkin	21	0	0	$\frac{1}{6}$	0	$-\frac{c}{2}$	0	$\lvert c\rvert\leqslant 1\sqrt{3}$ for stability
Wahlbin	1	0	$c\theta-(\operatorname{sgn} a)$	$\frac{1}{6}-c(\operatorname{sgn} a)\theta$	0	$-(\operatorname{sgn} a)$	0	
Dendy	2	0	$c\theta-(\operatorname{sgn} a)k'$	$\frac{1}{6}-c(\operatorname{sgn} a)k'\theta$	0	$-(\operatorname{sgn} a)k'$	0	$\operatorname{Re}[\Phi,k\Phi_x]\leqslant C_1\lvert\Phi\rvert^2, k=k'\Delta x,$ $C_1>0$
Raymond-Garder	3	$\frac{1}{2}$	$c\theta-(\operatorname{sgn} a)v$	$\frac{1}{6}-c(\operatorname{sgn} a)v\theta$	0	$-(\operatorname{sgn} a)v$	0	$v=1/\sqrt{15}$
penalty-Galerkin	7	all	$c\theta-(\operatorname{sgn} a)v_i$	$\frac{1}{6}-c(\operatorname{sgn} a)v_d\theta$	0	$-(\operatorname{sgn} a)v_d$	0	$v_i,v_d\approx C_x(1/\sqrt{15}), 0\leqslant C_x\leqslant 2$ for Euler equations
Hughes-Brooks SUPG	5					$-\left(\frac{a}{2}+\frac{k}{a\Delta x}\right)$		steady only, k = thermal conductivity
Euler-Petrov-Galerkin	6	0	0	$\frac{1}{6}-\frac{1}{6}(1-v)$	0	$\frac{(\operatorname{sgn} a)v}{2}$	0	$\lvert c\rvert\leqslant v\leqslant 1$
CN Petrov-Galerkin	6	$\frac{1}{2}$	$\frac{c}{2}-\frac{(\operatorname{sgn} a)v}{2}$	$\frac{1}{6}-\frac{v}{2}\left((\operatorname{sgn} a)\frac{c}{2}-\frac{1}{6}\right)$	0	$-\frac{(\operatorname{sgn} a)v}{2}$	0	$\lvert c\rvert\leqslant v\leqslant 1$
Warming-Beam explicit	23	0	0	0	0	$-\frac{c}{2}$	$-\frac{(1-c)}{2}$	$\lvert c\rvert\leqslant 1; a>0$
van Leer (MUSCL)	24	1	$c\theta$	$-c(\operatorname{sgn} a)\frac{\theta}{2}$	0	0	$-\frac{1}{2}$	Simple flux limiter ($\Phi^{\pm}\equiv 1$)
Hughes-Mallet	25	0	$-\frac{l(\operatorname{sgn} a)}{2}$	$\frac{1}{6}+\frac{1}{12}$	0	$-\frac{(\operatorname{sgn} a)}{2}$	0	l is flux limiter, $0\leqslant l\leqslant 1$

ORIGINAL PAGE IS
OF POOR QUALITY

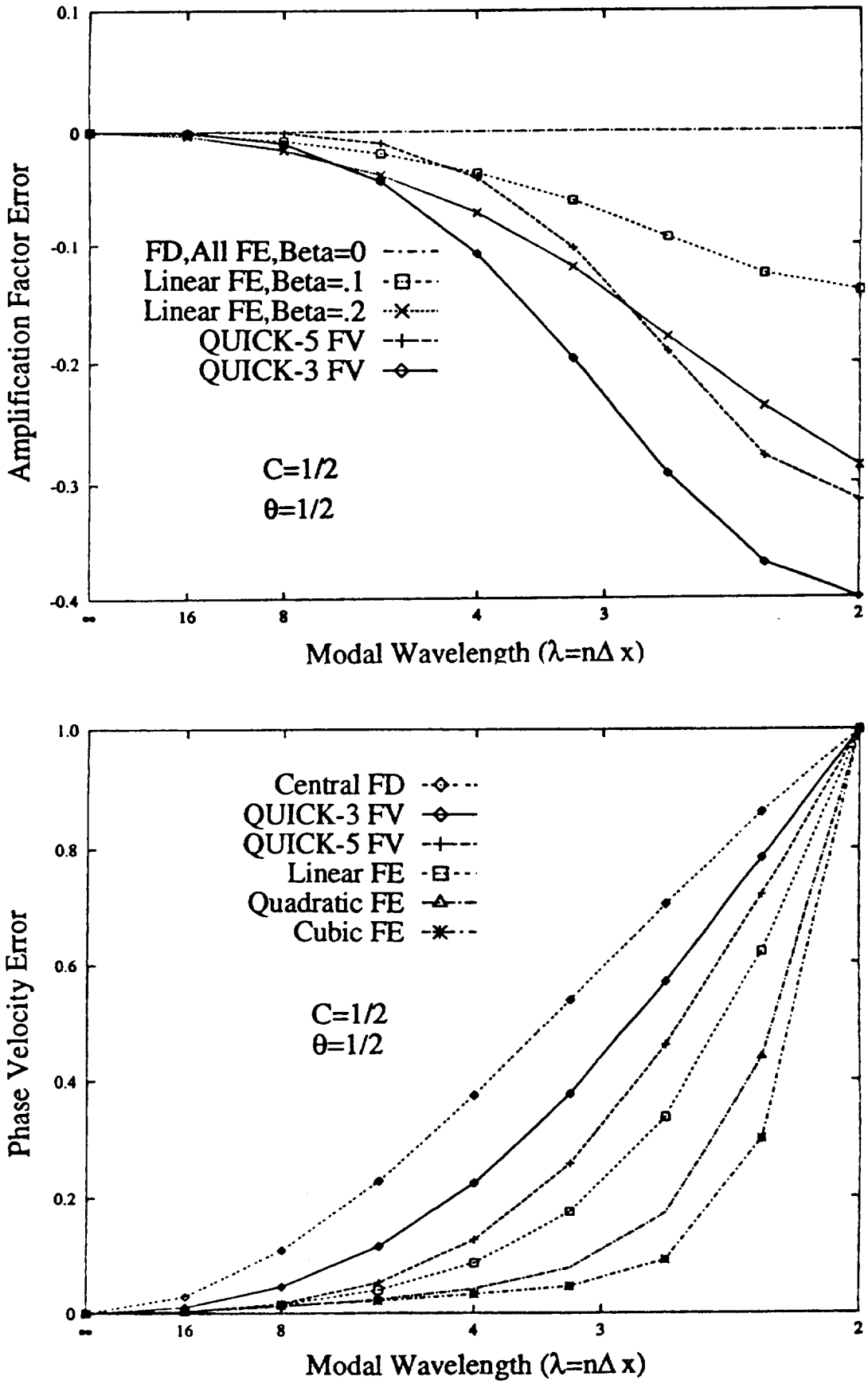


Figure 4.1 Amplification and phase velocity error distributions, various weak statement algorithms, Chaffin & Baker (1994)

5. The REMI AERO CFD ALGORITHM

5.1 Synopsis

The FE TWS^h CFD algorithm for aerodynamics applications has been derived and theoretically described. The next step is development of the actual matrix statements corresponding to the matrix algebra system (3.17). The resultant algorithm is coined "*REMI*," the acronym for the compressible state variable {Rho, Etot, Momentum $\underline{D}^T$ for $1 \le I \le d$.

The matrix formulation is expressed generically, then specialized to the use of the linear tensor product basis spanning quad and hex-shaped FE domains. Thereafter, the details of the construction are illustrated as "*template*" instruction sets for the *AKCESS.** software platform, which has now matured to replace the pilot code originally developed under the contract, c.f., Iannelli(1991). This section develops all pertinent details of the matrix statement construction.

5.2 Finite element TWS^h algorithm nomenclature

Equation (3.17) presents the homogeneous matrix expression $\{FQ\}$ of the TWS^h CFD algorithm construction. The formation of $\{FQ\}$ is via *assembly* over each finite element domain Ω_e of the mesh Ω^h, *i.e.*,

$$\{FQ\} = \underset{e}{S}\{FQ\}_e \tag{5.1}$$

$$\{FQ\}_e = [M]_e\left\{Q^p_{n+1} - Q_n\right\}_e + \Delta t\left(\theta\{RQ\}^p_{n+1} + (1-\theta)\{RQ_n\}\right) \tag{5.2}$$

The nodal discrete approximation state variable $\{Q\}_e$ is ordered as is the continuum, i.e., $q=\{\rho,E,m_i\}^T$, with discrete variable names $\{Q\}_e = \{\text{RHO,ETOT, M1,M2,M3}\}^T_e$. The appearance of velocity $u_i=m_i/\rho$ is discretely represented as the nodal ratio $\{MI/RHO\}_e$.The TWS^h definition at the generic element level is

$$TSW^h_e \equiv \int_{\Omega_e}\{N\}L^m(q^h)d\tau$$

$$= \int_{\Omega_e}\{N\}\left(L(q^h) - \beta(q)h\frac{\partial}{\partial x_j}\left(u_j\hat{u}_k\frac{\partial q}{\partial x_k}\right)_e\right)d\tau$$

$$= \int_{\Omega_e}\{N\}\left(\frac{\partial q^h}{\partial t} + \frac{\partial}{\partial x_j}\left(f_j - f^v_j\right)_e - s - \beta(q)h\frac{\partial}{\partial x_j}\left(u_j\hat{u}_k\frac{\partial q}{\partial x_k}\right)_e\right)d\tau \tag{5.3}$$

PRECEDING PAGE BLANK NOT FILMED

The substitution of (5.3) into the θ-implicit ODE algorithm, recall (3.14), and using (3.11), then yields the specific integral expressions for (5.2) as

$$[M]_e\left\{Q^p_{n+1}-Q_n\right\}_e=\int_{\Omega_e}\{N\}\{N\}^T\,\mathrm{d}\tau\left\{Q^p_{n+1}-Q_n\right\}_e \tag{5.4}$$

$$\begin{aligned}\{RQ\}_e=&-\int_{\Omega_e}\frac{\partial\{N\}}{\partial x_j}\left(f_j-f_j^v-\beta h u_j\hat{u}_k\frac{\partial q}{\partial x_k}\right)_e\mathrm{d}\tau\\&+\oint_{\partial\Omega_e}\{N\}\left(f_j-f_j^v-\beta h u_j\hat{u}_k\frac{\partial q}{\partial x_k}\right)_e\hat{n}_j\,\mathrm{d}\sigma\end{aligned} \tag{5.5}$$

Recalling (2.21), the elemental flux vector definitions for (5.5) are

$$f_j=\begin{Bmatrix}m_j\\ m_jm_i/\rho+\mathrm{Eu}p\delta_{ij}\\ (E+(\mathrm{Eu}/\mathrm{Ec})p)m_j/\rho\end{Bmatrix}_e,\quad f_j^v=\begin{Bmatrix}0\\ \mathrm{Re}^{-1}\sigma_{ij}\\ (\mathrm{ReEc})^{-1}\sigma_{ij}m_j/\rho-(\mathrm{Pe})^{-1}q_j\end{Bmatrix}_e \tag{5.6}$$

Finally, the solution parameters appearing in (5.6) are

$$p=\frac{\gamma-1}{\mathrm{Eu}}\left(\mathrm{Ec}E-m_jm_j/2\rho\right)_e \tag{5.7}$$

$$\sigma_{ij}=\mu(T)\left(\frac{\partial(m_i/\rho)}{\partial x_j}+\frac{\partial(m_j/\rho)}{\partial x_i}-\frac{2}{3}\frac{\partial(m_k/\rho)}{\partial x_k}\delta_{ij}\right)_e \tag{5.8}$$

$$q_j=-k(T)\left(\frac{\partial T}{\partial x_j}\right)_e \tag{5.9}$$

and all variables in (5.5)-(5.9) are expressed on Ω_e according to (3.11), *i.e.*,

$$q_e(\mathbf{x},t)\equiv\left\{N_k\left(\eta_j\right)\right\}^T\{Q(t)\}_e \tag{5.10}$$

The remaining construction in (3.17) is to derive the (Newton) jacobian $[JAC]$, which is also formed via *assembly* over the element-level constructions

$$[JAC]\equiv\underset{e}{S}[JAC]_e \tag{5.11}$$

and

$$[JAC]_e\equiv\frac{\partial\{FQ\}_e}{\partial\{Q\}_e}=[M]_e+\theta\Delta t\frac{\partial\{RQ\}_e}{\partial\{Q\}_e} \tag{5.12}$$

Upon forming (5.12), a quasi-Newton approximation constitutes elimination or substitution of terms therein, dependent on the specifics of the selected linear algebra procedure.

The construction of (5.4)-(5.5), hence also (5.12), requires integrals of products of FE bases, and their derivatives, formed on Ω_e. This operation is required done on every element, unless the mesh Ω^h is sufficiently regular such that a specific numerical quadrature is not required. This can be avoided, for efficiency, in certain instances, since an inline Gauss quadrature loop can add an order-of-magnitude or more to code execution time of the basic DO loops, cf., Baker and Pepper(1991, Ch. 6). (The *AKCESS.** platform contains both options, and each is "readible" from the identical *template* syntax construction.)

The formation of some element integrals involves handling of a coordinate transformation $\eta_j = \eta_j(x_i)$, since the FE tensor product basis functions are expressed in the local η_j (orthogonal) coordinate system. Figure 5.1 illustrates the essence, where superscript "hat" signifies the cartesian appearance of *every* (distorted) Ω_e in transform space. An element in physical space may possess curved edges/surfaces, and the necessary coordinate transformation is provided by the FE basis function interpolation of x_i on Ω_e. Hence, cf., Baker and Pepper (1991, p. 210), on *any* element

$$\left(x_i\right)_e = \left\{N\left(\eta_j\right)\right\}^T \{XI\}_e \ , \ 1 \le i, I \le d \tag{5.13}$$

where $\{XI\}_e$ contains the global (x_i) coordinates of the nodes of Ω_e.

Then, to form the matrix $[M]_e$ defined in (5.4), the calculus operation becomes

$$[M]_e = \int_{\Omega_e} \{N\}\{N\}^T det_e d\eta \tag{5.14}$$

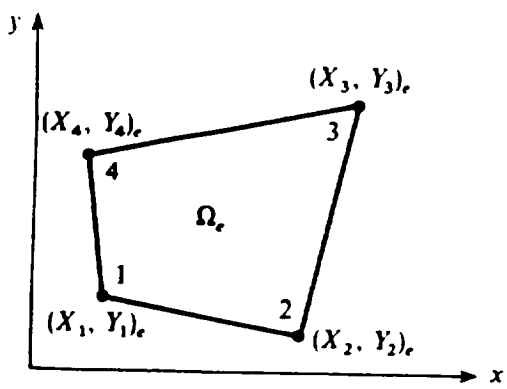
where det_e is the determinant of the jacobian of the forward transformation

$$det_e \equiv \det\left[\frac{\partial x_i}{\partial \eta_j}\right]_e \tag{5.15}$$

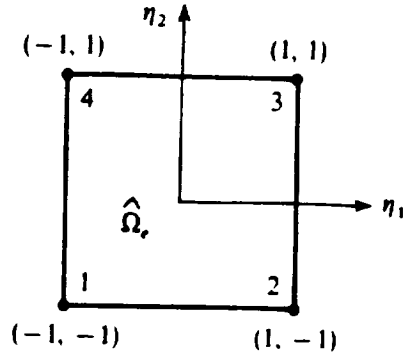
and $d\eta$ is the differential element equivalent of $d\tau$ in (5.4). Since (5.13) is non-linear, η_j-dependence remains present in (5.15), i.e., $det_e = det_e\left(\eta_j, XI_e\right)$.

The order of the element matrix (5.14) depends on FE basis polynomial degree (k) and the problem dimension d. The representative matrix element therein is

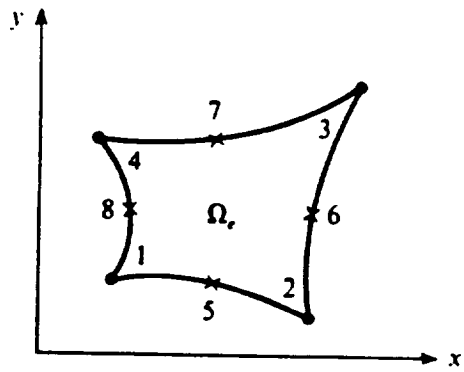
$$m_{\alpha,\beta} \equiv \int_{\hat{\Omega}_e} N_\alpha\left(\eta_j\right) N_\beta\left(\eta_j\right) det_e\left(\eta_j\right) d\eta \tag{5.16}$$



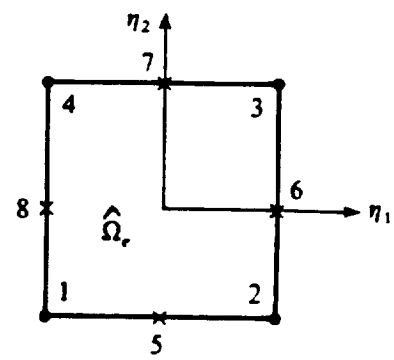
(a) Global reference frame



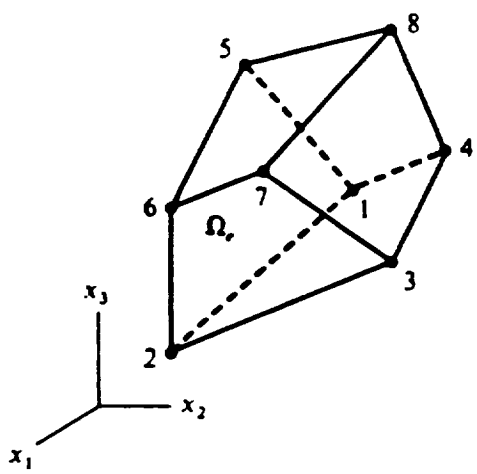
(b) Local reference frame



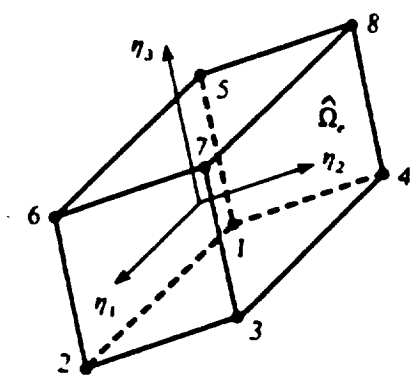
(a) Global reference frame



(b) Local reference frame



(a) Global reference frame



(b) Local reference frame

Figure 5.1 Finite element domains in physical space (Ω_e) and in transform space ($\hat{\Omega}_e$) for tensor product basis form.

where the indices range $1 \le (\alpha,\beta) \le (k+1)^d$, and (α,β) are row (column) indicators in $\{N\}$ ($\{N\}^T$). The integrand in (5.16) is a polynomial in η_j, the exact integral of which is evaluable via the Gauss quadrature replacement

$$m_{\alpha,\beta} = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} H_p H_q H_r \ \ integrand \left(\eta_j \Rightarrow (\eta_p \eta_q \eta_r\right) \tag{5.17}$$

where the integrand evaluation coordinate triples (η_p, η_q, η_r), and the Gaussian weights (H_p, H_q, H_r), depend on the degrees (P,Q,R) of the quadrature rule. Figure 5.2 illustrates the d=2 case for the symmetric implementation (P=Q) for $1<P<3$.

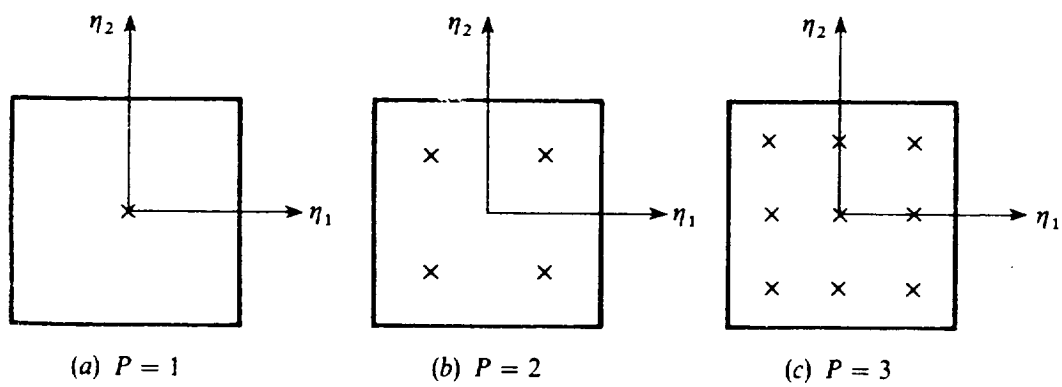


Figure 5.2 Gauss symmetric quadrature coordinates for d=2.

The selection of P is governed by the highest degree in the polynomial integrand; an exact (symmetric) evaluation results when $2P+1$ equals or exceeds this extremum degree. Table 5.1 lists the corresponding Gauss point coordinates and associated weights.

Table 5.1 Gauss quadrature coordinates and weights, d=2.

Order (P)	Coordinate η_1^p	Coordinate η_2^q	Weight H_p	Weight H_q
1	0.0	0.0	1.0	1.0
2	$\pm 1\sqrt{3}$	$\pm 1\sqrt{3}$	1.0	1.0
3	0.0	0.0	8/9	8/9
	$\pm\sqrt{0.6}$	$\pm\sqrt{0.6}$	5/9	5/9

Integration thus being completed, the element mass matrix $[M]_e$ becomes the array of numbers $m_{\alpha,\beta}$ that depend on the specific element Ω_e. Notationally then,

$$[M]_e = \int_{\hat{\Omega}_e} \{N\}\{N\}^T \, det_e \, d\eta \equiv [\text{M200E}] \tag{5.18}$$

where [M200E] is the generic *variable name* given to the array $m_{\alpha,\beta}$ on Ω_e. The prefix "M" stands for (element) Matrix, and becomes replaced by (A,B,C) for d=1,2,3. The "2" signifies that two basis functions $\{N\}$ reside in the integral, and the succeeding two zeroes indicate neither basis is differentiated. Finally, the suffix "E" denotes the matrix is indeed Element-dependent data.

In the event that det_e in (5.16) is nominally uniform on Ω_e, as occurs *exactly* for *any* parallelogram finite element (independent of included angle variation), only minor interpolation error is introduced by extracting it from the integrand as an average. Then, the remaining integrand is *element-independent*, hence quadrature is performed once only for *all* elements in Ω^h. Notationally, (5.18) is then replaced as

$$[M]_e \equiv \overline{det}_e \int_{\hat{\Omega}_e} \{N\}\{N\}^T \, d\eta \equiv DET_e[\text{M200}] \tag{5.19}$$

and [M200] is a universal matrix of numbers representing the integral of two non-differentiated basis functions on d=1,2,3 dimensions for M⇒(A,B,C). Further DET_e is proportional to length/area/volume of Ω_e and Table 5.2 presents the k=1 basis data

Table 5.2 FE k=1 basis interpolation matrix [M200] for d=1,2,3

dimension d	DET_e	[A200]	[B200]	[C200]
1	$\frac{1}{2}l_e$	$1/3\begin{bmatrix}2 & 1\\ 1 & 2\end{bmatrix}$		
2	$\frac{1}{4}A_e$		$\frac{1}{9}\begin{bmatrix}4&2&1&2\\2&4&2&1\\1&2&4&2\\2&1&2&4\end{bmatrix}$	
3	$\frac{1}{8}V_e$			$\frac{1}{27}\begin{bmatrix}8&4&2&4&4&2&1&2\\4&8&4&2&2&4&2&1\\2&4&8&4&1&2&4&2\\4&2&4&8&2&1&2&4\\4&2&1&2&8&4&2&4\\2&4&2&1&4&8&4&2\\1&2&4&2&2&4&8&4\\2&1&2&4&4&2&4&8\end{bmatrix}$

The remaining integrals in $\{FQ\}_e$, (5.5), are contributions from the "residual" $\{RQ\}_e$, (5.3). Evaluating all such contributions involves the coordinate transformation (5.13) via the chain rule, *i.e.*,

$$\{RQ\}_e = -\int_{\hat{\Omega}_e} \frac{\partial\{N\}}{\partial\eta_i}\left(\frac{\partial\eta_i}{\partial x_j}\right)_e \left(f_j - f_j^v - \beta^h u_j \hat{u}_k \frac{\partial q}{\partial x_k}\right)_e det_e \, d\eta \tag{5.20}$$

By definition, the required inverse coordinate transformation matrix is

$$\left[\frac{\partial\eta_i}{\partial x_j}\right]_e = \left[\frac{\partial x_i}{\partial\eta_j}\right]_e^{-1} = \frac{1}{det_e}[cofactor\,(i,j)]_e \tag{5.21}$$

where [cofactor $(i,j)]_e$ is the matrix of signed minors, transformed, of the forward transformation on Ω_e. The inverse det_e in (5.21) cancels the like term in (5.20), and labeling the cofactor matrix entries as the array "ETIJ_e" *i.e.*, "eta_{ij}" then the first integrand term becomes

$$\{RQ\}_e = -\int_{\hat{\Omega}_e} \mathrm{ETI}J_e \frac{\partial\{N\}}{\partial\eta_i}(f_j)_e \, d\eta \tag{5.22}$$

For example, for q=ρ, then from (2.1) $f_j(q=\rho)=m_j$, hence

$$\{\mathrm{RR(f)}\}_e = -\int_{\hat{\Omega}_e} \mathrm{ETI}J_e \frac{\partial\{N\}}{\partial\eta_i}\{N\}^T d\eta \{MI\}_e \tag{5.23}$$

Note the tensor indices are fully contracted in (5.23), hence $\{\mathrm{RR}\}_e$ is indeed a single array (column matrix). The integrand in (5.23) is an order $(k+1)^d$ matrix, with each term therein the sum of $1<(i,I,j,J)< d$ terms. Gaussian quadrature applied to each of these expressions completes the integrations, hence (5.23) becomes, notationally

$$\{\mathrm{RR(f)}\}_e = -[\mathrm{M2}J0\mathrm{E}]\{MI\}_e \tag{5.24}$$

where $1 < I < d$ is the sole remaining tensor index.

The "E" suffix in (5.24) indicates the data entries are element-dependent via the η_i-dependence in the transformation matrix ETIJ_e. These data are products of differences in nodal coordinates $\{XI\}_e$, which reduce to element constants for *any* parallelogram Ω_e. Hence, committing usually modest interpolation error, the simplified counterpart of (5.19) for (5.24) is

$$\{\mathrm{RR(f)}\}_e \cong -\mathrm{ETI}J_e\left[[\mathrm{M2}J0]\{MJ\}_e\right] \tag{5.25}$$

which also has full contraction of tensor indices. Now, [M2JO] is *element-independent* data, hence applicable on every Ω_e that is sufficiently close to a parallelogram, c.f., Baker & Pepper (1991, Ch. 6).

The second contribution to the $\{RQ\}_e$ array is from the viscous flux vector f_j^v, which contains derivatives itself, recall (5.8)-(5.9). Therefore, two coordinate

transformations are involved in the integrand, leading to the representative contribution, e.g., for $q=E$,

$$\left\{\mathrm{RE}\left(\mathbf{f}^{v}\right)\right\}_e = \int_{\Omega_e} \frac{\partial\{N\}}{\partial x_j}\frac{1}{\mathrm{Pe}}k\frac{\partial\{N\}^T}{\partial x_j}\mathrm{d}\tau\{\mathrm{TEMP}\}_e$$
$$= \frac{1}{\mathrm{Pe}}\int_{\hat{\Omega}_e} \frac{\partial\{N\}}{\partial \eta_i}\left(\frac{\partial \eta_i}{x_j}\right)_e k\left(\eta_i\right)\frac{\partial\{N\}^T}{\partial \eta_k}\left(\frac{\partial \eta_k}{\partial \eta_j}\right)_e det_e\,\mathrm{d}\eta\{\mathrm{TEMP}\}_e \tag{5.26}$$

In (5.26), det_e in the numerator is canceled as before, however a det_e now remains in the denominator. Therefore, the integrand in (5.26) is a rational polynomial in η_j, with the added complexity of a variable diffusion coefficient $k(\eta_j)_e$. Gauss quadrature remains applicable, however, hence notationally (5.26) becomes

$$\left\{\mathrm{RE}\left(\mathbf{f}^{v}\right)\right\}_e = \frac{1}{\mathrm{Pe}}[\mathrm{M2}KK]_e\{\mathrm{TEMP}\}_e \tag{5.27}$$

where $[\mathrm{M2}KK]_e$ is element-dependent data, and the range of the repeated tensor index pair is $1 < K < d$, in accord with the transformed *laplacian* equivalence.

The computational cost of guassian quadrature in forming (5.27) on every element is quite substantial, hence a simplified form is again sought. For nominal parallelogram domains Ω_e, the transformation metric data in (5.26) are essentially uniform constants. Hence, minor interpolation error can result upon their extraction, leading to

$$\left\{\mathrm{RE}\left(\mathbf{f}^{v}\right)\right\}_e \cong \frac{1}{\mathrm{Pe}}\mathrm{ET}IJ_e\mathrm{ET}KJ_e\mathrm{DET}_e^{-1}\int_{\hat{\Omega}_e} k_e\frac{\partial\{N\}\,\partial\{N\}^T}{\partial \eta_i\,\partial \eta_k}\mathrm{d}\eta\{\mathrm{TEMP}\}_e$$
$$= \frac{1}{\mathrm{Pe}}\mathrm{ET}IJ_e\mathrm{ET}KJ_e\mathrm{DET}_e^{-1}\{\mathrm{COND}\}_e^T[\mathrm{M30}IK]\{\mathrm{TEMP}\}_e \tag{5.28}$$

Equation (5.28) remains fully contracted over tensor index pairs, and the triple summation loop on I,J,K, required to form *each* entry, is clearly defined. The metric data $\mathrm{ET}IJ_e$ is a dxd array, $\{\mathrm{COND}\}_e^T$ is an element-rank row matrix of nodal conductivity, and $[\mathrm{M30}IK]$ is an *element-independent*, degree-one *hyper*matrix interpolating distributed data in concert with two derivative operators, hence the "3." Thereafter, "0" indicates FE (non-differentiated) basis interpolation of k_e, and "IK" denote all derivative combinations corresponding to the transformed laplacian, with range $1 < (I, K) < d$.

This completes the introduction to notational essence for the *REMI* FE CFD algorithm. The formulation can now be concisely stated for arbitrary problem dimension d, whereafter element matrices with prefix "M" become "A,B,C" for the specific situation. In the following, the distinction between gaussian quadrature and averaged-metric data construction is highlighted, and *AKCESS.** contains both options via the "E" suffix. Hence, if numerical integration is required, then the hypermatrix [M30*IK*] is replaced by ⇒[M30*IK*E], and the tensor index DO loop remains clearly indicated.

5.3. *REMI* algorithm matrix statement illustrations

This section illustrates construction details of the *REMI* algorithm matrix statement, (5.2), that becomes assembled, (5.1), to the form the right side of a linear algebra procedure. For $\alpha = 0 = \gamma$ in the TWS^h formulation, then the lead term (5.4) is uniform across all members $\{Q\}_e$ of the discrete state variable. Hence,

$$[M]_e\left\{Q^p_{n+1} - Q_n\right\}_e = [\mathrm{M200}]_e\left\{Q^p_{n+1} - Q_n\right\}_e$$

$$= [\mathrm{M200}]_e\left(\begin{Bmatrix}\mathrm{RHO}\\ \mathrm{ETOT}\\ \mathrm{M1}\\ \mathrm{M2}\\ \mathrm{M3}\end{Bmatrix}^p_{e,n+1} - \begin{Bmatrix}\mathrm{RHO}\\ \mathrm{ETOT}\\ \mathrm{M1}\\ \mathrm{M2}\\ \mathrm{M3}\end{Bmatrix}_{e,n}\right) \tag{5.29}$$

The "residual" terms, (5.5), typically contain convection, diffusion and dissipation contributions, with element matrices generically labeled "$[V]_e$, $[D]_e$ and $[DB]_e$" and companion surface integrals with prefix "*S*". Then, for example, for state variable member ρ, and viewing (5.5), (5.6)

$$\{\mathrm{RR}\}_e = -[V]_e + [DB]_e\{\mathrm{RHO}\}_e - [SDB]_e\{\mathrm{RHO}\}_e$$

$$= -[\mathrm{M2}I0]_e\{\mathrm{M}I\}_e + \beta_R\hat{\mathrm{U}}K_e\{\mathrm{U}J\}^T_e[\mathrm{M30}JK]_e\{\mathrm{RHO}\}_e \tag{5.30}$$

$$+[\mathrm{S200}]_e\{\mathrm{M}I\mathrm{N}I\}_e - \beta_R\hat{\mathrm{U}}K_e\{\mathrm{U}J\mathrm{N}J\}^T_e[\mathrm{S300}K]_e\{\mathrm{RHO}\}_e$$

In (5.30), the tensor indices contract fully, as they must, and in the TWS^h β-dissipation term, the decision is made to element-average the velocity unit vector $\hat{u}_k$ and interpolate the convection velocity u_j. In the last two terms $\{\mathrm{M}I\mathrm{N}I\}_e$, and $\{\mathrm{U}J\mathrm{N}J\}_e$ contain the nodal values of the dot products $m_i\hat{n}_i$ and $u_j n_j$. [S200] is the interpolation surface matrix while [S3000*R*] is the corresponding β-term surface integral matrix, both created by use of the Green-Gauss divergence theorem in WS^N, recall (3.9).

Every matrix in (5.30 is element-dependent, as denoted by subscript "*e*". In practice, each can be code implemented via the operation (5.27)-(5.28) in an *AKCESS.**

template instruction set. Viewing (5.28), every element-level contribution to the residual (and the time) term is constituted of six types of data, i.e.,

$$\{FQ\}_e = (\text{data})(\text{data})_e \{\text{data}\}_e^T (\text{metrics, det})_e [\textit{hyper}\text{matrix}]\{Q \text{ or data}\}_e \tag{5.31}$$

where:

(data)	=	global constants
$(\text{data})_e$	=	element average data
$\{\text{data}\}_e$	=	element distributed data
$(\text{metrics, det})_e$	=	$ETIJ_e$ and exponent on DET_e
[*hyper*matrix]	=	FE matrix or *hyper*matrix
$\{Q \text{ or data}\}_e$	=	FE matrix post multiplier

The "long hand" form of the first right side term in (5.30) for d=2 is

$$\begin{aligned} &-DET_e^0(ET11_e[M210]+ET12_e[M220])\{M1\}_e \\ &+\left(ET21_e[M210]+ET22_e\ [M220]\right)\{M2\}_e \end{aligned} \tag{5.32}$$

The *AKCESS.** storage arrangement for the d x d metric data set is a d x 1 column array, *i.e.*, with locations

$$d=2\text{: } ETIJ_e \Rightarrow \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}; \quad d=3\text{: } ETIJ_e \Rightarrow \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

Therefore, the d=2 *template* syntax instruction set for (5.32) is,

$$\begin{aligned} \{RR(f)\}_e &= (-)(\ )(\ )(102;0)(B210)(M1) \\ &+(-)(\ )(\ )(304;0)(B220)(M2) \end{aligned} \tag{5.33}$$

An empty parenthesis defaults to unity; the (-) entry equals negative one. The "0" separating metric data entries is interpreted as a plus sign, while the "0" following the semi-colon is the exponent on DET_e. The d=3 template expression is

$$\begin{aligned} \{RR(f)\}_e &= (-)(\ )(\ )(10203;0)(C210)(M1) \\ &+(-)(\ )(\ )(40506;0)(C220)(M2) \\ &+(-)(\ )(\ )(70809;0)(C230)(M3) \end{aligned} \tag{5.34}$$

Comparing (5.30) to (5.34), the compact notation in the former contains several detailed terms. This is more evident when tensor summation indices are involved in the residual expression, e.g., the second term in (5.30). However *template* syntax clearly defines every term in such summations. Figure 5.3 illustrates the complete TWS^h algorithm statement for $\{FR\}_e$ in d=2. The line below the command line "RESIDUALS," and the variable definition "RHO 1..." contains (5.33), and the next 16 lines constitute

the β-dissipation term involving [M30*JK*] in (5.30). The common data in these instructions include:

(HBM)	:	a global on-off switch for dissipation
(.,UMGB)	:	equal to $\beta/\|u\|$, where β is the dissipation coefficient and $\|u\|^{-1}$ normalizes u_k to $\hat{u}_k$
(...;-0.5)	:	the net exponent on DET_e, for length scale h in (5.5) equated to $(\mathrm{DET}_e)^{1/d}$

The last two (surface) matrix expressions in (5.30) become *template* instructions on no-flow/inflow/outflow surfaces, according to several options. On a no-flow surface, $m_i n_i$ hence $u_j n_j$ vanishes identically, so no evaluation results, whereupon the TWS^h algorithm automatically enforces (in the weak sense) the inviscid flow tangency boundary condition. Since $m_i \hat{n}_i$ never vanishes at inflow/outflow, and the actual distribution is (almost) never *given* data (recall Table 2.1), the *AKCESS.** remedy is to "undo" the integration by parts that created the $[\mathrm{S200}]_e$ term. Looking back to (5.3) and (5.5), this is easily accomplished by revising (5.22)-(5.23), the definition of $\{\mathrm{RR(f)}\}_e$, to

$$\begin{aligned}\{\mathrm{RR(f)}\}_e &= \int_{\Omega_e} \{N\}\frac{\partial m_j}{\partial x_j} d\tau = \int_{\hat{\Omega}_e} \{N\}\mathrm{ETIJ}_e \frac{\partial \{N\}^T}{\partial \eta_i} d\eta \{MJ\}_e \\ &= [\mathrm{M20}JE]_e \{\mathrm{M}J\}_e = \mathrm{ETIJ}_e [\mathrm{M201}]\{\mathrm{M}J\}_e\end{aligned} \tag{5.35}$$

Figure 5.3 confirms the *template* of this rearrangement, as the first few lines on inflow (set 3) and outflow (set 4) boundaries.

The surface integral created by the tensor β-dissipation term involves a directional derivative contracted with velocity unit vector $\hat{u}_k$. This also is typically not *given* data on any through-flow boundary, although the assumption of vanishing normal derivative is often implied. Numerical experimentation, discussed in Section 6, confirms that replacement of the tensor β diffusivity form (4.23) with the *positivity-ensured* form (4.24) better admits *weak* enforcement of vanishing normal derivatives. The second matrix term in (5.30) can then be replaced as

$$[DB]_e \{\mathrm{RHO}\}_e = \beta_R \{\mathrm{UMAG}\}_e^T [\mathrm{M30}JJ]_e \{\mathrm{RHO}\}_e \tag{5.36}$$

The companion surface integral $[\mathrm{S30NN}]_e$ is not formed, and $\{\mathrm{UMAG}\}_e$ in (5.18) is the nodal distribution of $|\mathbf{v}|$ as computed via (4.24). The *template* in Fig. 5.3 illustrates the

```
TITLE        **** TEMPLATE FILE TEMP.CNS2D.REMI ****
  CNS2D TWS ALGORITHM, TENSOR MATRIX A JACOBIANS (12/23/93)
RESIDUALS
 RHO  1 # VARIABLE, SET NO., --- [M] * {[RHO.NEW] - [RHO.OLD]}
() () () (;1) (B200) (-RHO)
 RHO  2 # VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
 () () () (102;0)  (B201) (M1) + () () () (304;0) (B202) (M2)
+(HBR)(U1,UMGB)(U1)(11;-0.5)(B3011)(RHO)
+(HBR)(U1,UMGB)(U2)(12;-0.5)(B3011)(RHO)
+(HBR)(U2,UMGB)(U1)(21;-0.5)(B3011)(RHO)
+(HBR)(U2,UMGB)(U2)(22;-0.5)(B3011)(RHO)
+(HBR)(U1,UMGB)(U1)(13;-0.5)(B3012)(RHO)
+(HBR)(U1,UMGB)(U2)(14;-0.5)(B3012)(RHO)
+(HBR)(U2,UMGB)(U1)(32;-0.5)(B3012)(RHO)
+(HBR)(U2,UMGB)(U2)(24;-0.5)(B3012)(RHO)
+(HBR)(U1,UMGB)(U1)(31;-0.5)(B3021)(RHO)
+(HBR)(U1,UMGB)(U2)(32;-0.5)(B3021)(RHO)
+(HBR)(U2,UMGB)(U1)(41;-0.5)(B3021)(RHO)
+(HBR)(U2,UMGB)(U2)(42;-0.5)(B3021)(RHO)
+(HBR)(U1,UMGB)(U1)(33;-0.5)(B3022)(RHO)
+(HBR)(U1,UMGB)(U2)(34;-0.5)(B3022)(RHO)
+(HBR)(U2,UMGB)(U1)(43;-0.5)(B3022)(RHO)
+(HBR)(U2,UMGB)(U2)(44;-0.5)(B3022)(RHO)
 RHO  3 # VARIABLE, SET NO., --- INFLOW BOUNDARY SET FOR {RQ}
+(PHRI)(U1,UMGB)(U1)(11;-0.5)(B3011)(RHO)
+(PHRI)(U1,UMGB)(U2)(12;-0.5)(B3011)(RHO)
+(PHRI)(U2,UMGB)(U1)(21;-0.5)(B3011)(RHO)
+(PHRI)(U2,UMGB)(U2)(22;-0.5)(B3011)(RHO)
+(PHRI)(U1,UMGB)(U1)(13;-0.5)(B3012)(RHO)
+(PHRI)(U1,UMGB)(U2)(14;-0.5)(B3012)(RHO)
+(PHRI)(U2,UMGB)(U1)(32;-0.5)(B3012)(RHO)
+(PHRI)(U2,UMGB)(U2)(24;-0.5)(B3012)(RHO)
+(PHRI)(U1,UMGB)(U1)(31;-0.5)(B3021)(RHO)
+(PHRI)(U1,UMGB)(U2)(32;-0.5)(B3021)(RHO)
+(PHRI)(U2,UMGB)(U1)(41;-0.5)(B3021)(RHO)
+(PHRI)(U2,UMGB)(U2)(42;-0.5)(B3021)(RHO)
+(PHRI)(U1,UMGB)(U1)(33;-0.5)(B3022)(RHO)
+(PHRI)(U1,UMGB)(U2)(34;-0.5)(B3022)(RHO)
+(PHRI)(U2,UMGB)(U1)(43;-0.5)(B3022)(RHO)
+(PHRI)(U2,UMGB)(U2)(44;-0.5)(B3022)(RHO)
 RHO  4 # VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
+(PHRO)(U1,UMGB)(U1)(11;-0.5)(B3011)(RHO)
+(PHRO)(U1,UMGB)(U2)(12;-0.5)(B3011)(RHO)
+(PHRO)(U2,UMGB)(U1)(21;-0.5)(B3011)(RHO)
+(PHRO)(U2,UMGB)(U2)(22;-0.5)(B3011)(RHO)
+(PHRO)(U1,UMGB)(U1)(13;-0.5)(B3012)(RHO)
+(PHRO)(U1,UMGB)(U2)(14;-0.5)(B3012)(RHO)
+(PHRO)(U2,UMGB)(U1)(32;-0.5)(B3012)(RHO)
+(PHRO)(U2,UMGB)(U2)(24;-0.5)(B3012)(RHO)
+(PHRO)(U1,UMGB)(U1)(31;-0.5)(B3021)(RHO)
+(PHRO)(U1,UMGB)(U2)(32;-0.5)(B3021)(RHO)
+(PHRO)(U2,UMGB)(U1)(41;-0.5)(B3021)(RHO)
+(PHRO)(U2,UMGB)(U2)(42;-0.5)(B3021)(RHO)
+(PHRO)(U1,UMGB)(U1)(33;-0.5)(B3022)(RHO)
+(PHRO)(U1,UMGB)(U2)(34;-0.5)(B3022)(RHO)
+(PHRO)(U2,UMGB)(U1)(43;-0.5)(B3022)(RHO)
+(PHRO)(U2,UMGB)(U2)(44;-0.5)(B3022)(RHO)
```

Figure 5.3 *AKCESS*.AERO *REMI* template for $\{FR\}_e$

removal of the (5.30) term and replacement with (5.18) on inflow and outflow boundaries. Therein, (PHBI) and (PHBO) are "switches" defined in the "SCALARS" array to facilitate decisions on use.

The convection terms in TWS^h for state variable members m_i and E are non-linear, hence algorithm and *AKCESS.** manipulations are similarly appropriate to account for TWS^h-created surface integrals. For example, for axial momentum the residual contains the terms

$$\begin{aligned}\{\mathrm{RM1}\}_e = &-[V]_e\{M1\}_e + [D]_e\{M1\}_e + [DB]_e\{M1\}_e \\ &+ [S]_e\{M1\}_e - [SD]_e\{M1\}_e - [SDB]_e\{M1\}_e\end{aligned} \tag{5.37}$$

Viewing (5.5), (5.8), the FE matrix syntax for (5.37) is

$$\begin{aligned}\{\mathrm{RM1}\}_e = &-\{\mathrm{U}J\}_e^T[\mathrm{M30}JO]_e\{\mathrm{M1}\}_e + \mathrm{Eu}[\mathrm{M201}]_e\{\mathrm{P}\}_e \\ &+ \mathrm{Re}^{-1}\mathrm{MU}_e\Big([\mathrm{M2}JJ]_e\{\mathrm{U1}\}_e + [\mathrm{M2}J1]_e\{UJ\}_e - \tfrac{2}{3}[\mathrm{M21K}]_e\{\mathrm{UK}\}_e\Big) \\ &+ \beta_1\hat{\mathrm{U}}K_e\{\mathrm{U}J\}_e^T[\mathrm{M30}JK]_e\{\mathrm{M1}\}_e \\ &- \{\mathrm{U}J\mathrm{N}J\}_e^T[\mathrm{S3000}]_e\{\mathrm{M1}\}_e - [\mathrm{SDB}]_e\{\mathrm{M1}\}_e \\ &- \mathrm{Re}^{-1}\mathrm{MU}_e\Big([\mathrm{S20N}]_e\{\mathrm{U1}\}_e + \tfrac{1}{3}[\mathrm{S21N}]_e\{\mathrm{UN}\}_e\Big)\end{aligned} \tag{5.38}$$

Comparing the form of (5.38) to (5.5), note that the pressure term is not subjected to the Green-Gauss divergence operation, that (non-D) viscosity is element-averaged rather than distributed, and since J and K are repeated indices in the Re^{-1} term, a simplification occurs. For the application of (5.38) to a turbulent flow, the accounting for the large variation in turbulent eddy viscosity μ^t, recall (2.16), is readily accomplished via alteration of the diffusion terms in (5.38) to *hyper*matrix form

$$[D]_e[M1]_e = \mathrm{Re}^{-1}\{\mathrm{MUT}\}_e^T\Big([M30JJ]_e\{\mathrm{N1}\}_e + \tfrac{1}{3}[\mathrm{M30}J1]\{\mathrm{U}J\}_e\Big) \tag{5.39}$$

where $\{\mathrm{MUT}\}_e$ contains nodal μ^t on Ω_e, and a companion change occurs to $[SD]_e\,\{M1\}_e$.

Various rearrangements to (5.38)-(5.39) are required for constraints created by no-flow and through-flow boundaries. As with $\{\mathrm{RR}\}_e$, (5.30), the inviscid tangency boundary condition is intrinsic to (5.38) with no evaluation of the [S3000] term. Similarly, at inflow and outflow boundaries, rather than estimate the missing data $\{\mathrm{U}J\mathrm{N}J\}_e$, the divergence thereon is "undone" yielding the chain rule replacement

$$\begin{aligned}-[V]_e\{M1\}_e &\Rightarrow \{\mathrm{U}J\}_e^T[\mathrm{M300}J]_e\{\mathrm{M1}\}_e + \{\mathrm{M1}\}_e^T[\mathrm{M300}J]_e\{\mathrm{U}J\}_e \\ &= \{\mathrm{U}J\}_e^T\big([\mathrm{M300}J]_e + [\mathrm{M3}J00]_e\big)\{\mathrm{M1}\}_e\end{aligned} \tag{5.40}$$

which illustrates differentiation index manipulations to simplify the variable groupings. The discussed variation on the β-terms $[DB]_e$ and $[SDB]_e$ on through-flow boundaries is employed on all state variable residuals.

The remaining surface integral $[SD]_e$ is typically assumed to vanish on through-flow boundaries, and admits the specific implementation of a *drag* boundary condition on an impervious surface. Returning to the continuum form, (5.5)-(5.8), the weak statement term is

$$\begin{aligned}\int_{\Omega_e}\{N\}\frac{\partial}{\partial x_j}\left(f_j\right)d\tau &= -\frac{1}{\mathrm{Re}}\int_{\Omega_e}\{N\}\frac{\partial}{\partial x_j}\left(\mu\frac{\partial u_i}{\partial x_j}+\cdots\right)d\tau \\ &= \frac{1}{\mathrm{Re}}\int_{\Omega_e}\frac{\partial\{N\}}{\partial x_j}\mu\frac{\partial u_i}{\partial x_j}d\tau-\frac{1}{\mathrm{Re}}\oint_{\partial\Omega_e}\{N\}\mu\frac{\partial u_i}{\partial x_j}\hat{n}_j\,d\sigma\end{aligned} \tag{5.41}$$

The surface integral term clearly generalizes to $\sigma_{ij}\hat{n}_j$ which is the shear stress on a boundary surface segment of $\partial\Omega$, the boundary of Ω. Hence, in (5.37), for a slip wall with drag, the replacement expression is

$$-[SD]_e\{M1\}_e \Rightarrow -\mathrm{Re}^{-1}[\mathrm{S200}]_e\{\mathrm{TAU1N}\}_e \tag{5.42}$$

where $\{\mathrm{TAU1N}\}_e$ contains the applied shear stress relationship for m_1. Figure 5.4 illustrates excerpts from the *AKCESS.* template* for (5.37), with modifications, for d=2.

5.4 The *REMI* RaNS/E algorithm

With notational preliminaries completed, the *REMI* FE TWS^h algorithm for RaNS/E aerodynamics applications is now detailed. The development herein uses hybrid compact notation; the *AKCESS.* template* with full detail for d=2 is included in Appendix A.

The generic element level TWS^h matrix statement is

$$\{FQ\}_e^p=[M]_e\left\{Q_{n+1}^p-Q_n\right\}_e+\Delta t\left(\theta\{RQ\}_{n+1}^p+\{RQ\}_n\right)_e \tag{5.43}$$

with contributions

$$[M]_e\left\{Q_{n+1}^p-Q_n\right\}_e=\int_{\Omega e}\{N\}\{N\}^T d\tau\left\{Q_{n+1}^p-Q_n\right\}_e \tag{5.4}$$

$$\begin{aligned}\{RQ\}_e &= -\int_{\Omega_e}\frac{\partial\{N\}}{\partial x_j}\left(f_j-f_j^v-\beta h u_j\hat{u}_k\frac{\partial q}{\partial x_k}\right)_e d\tau \\ &\quad +\oint_{\partial\Omega_e}\{N\}\left(f_j-f_j^v-\beta h u_j\hat{u}_k\frac{\partial q}{\partial x_k}\right)_e\hat{n}_j\,d\sigma\end{aligned} \tag{5.5}$$

```
TITLE          **** TEMPLATE FILE TEMP.CNS2D.REMI ****
  CNS2D  TWS ALGORITHM,  TENSOR MATRIX A JACOBIANS (12/23/93)

RESIDUALS
 M1   1  # VARIABLE,  SET NO.,  ---  [M] * {[M1.NEW] - [M1.OLD]}
 ()()()(;1)(B200)(-M1)
 M1   2  # VARIABLE,  SET NO.,  ---  {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
(-)()(U1+U2) (102;0) (B3010) (M1) + (-) () (U1+U2)  (304;0) (B3020)  (M1)
+(EULER) () () (1;0)  (B201)  (PRSC) + (EULER) () () (3;0) (B202) (PRSC)
+(PDUM2,REI)()()(1122;-1)(B211)(U1)+(PDUM2,REI)()()(3344;-1)(B222)(U1)
+(PDUM2,REI)()()(1324;-1)(B221)(U1)+(PDUM2,REI)()()(1324;-1)(B212)(U1)
+(HBR)(U1,UMGB)(U1)(11;-0.5)(B3011)(M1)
+(HBR)(U1,UMGB)(U2)(12;-0.5)(B3011)(M1)
+(HBR)(U2,UMGB)(U1)(21;-0.5)(B3011)(M1)
+(HBR)(U2,UMGB)(U2)(22;-0.5)(B3011)(M1)
+(HBR)(U1,UMGB)(U1)(13;-0.5)(B3012)(M1)
+(HBR)(U1,UMGB)(U2)(14;-0.5)(B3012)(M1)
+(HBR)(U2,UMGB)(U1)(32;-0.5)(B3012)(M1)
+(HBR)(U2,UMGB)(U2)(24;-0.5)(B3012)(M1)
+(HBR)(U1,UMGB)(U1)(31;-0.5)(B3021)(M1)
+(HBR)(U1,UMGB)(U2)(32;-0.5)(B3021)(M1)
+(HBR)(U2,UMGB)(U1)(41;-0.5)(B3021)(M1)
+(HBR)(U2,UMGB)(U2)(42;-0.5)(B3021)(M1)
+(HBR)(U1,UMGB)(U1)(33;-0.5)(B3022)(M1)
+(HBR)(U1,UMGB)(U2)(34;-0.5)(B3022)(M1)
+(HBR)(U2,UMGB)(U1)(43;-0.5)(B3022)(M1)
+(HBR)(U2,UMGB)(U2)(44;-0.5)(B3022)(M1)
 M1   3  # VARIABLE, SET NO., --- INFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(M1)+()()(U1+U2)(304;0)(B3020)(M1)
+()()(U1+U2)(102;0)(B31P1)(M1)+()()(U1+U2)(304;0)(B32P2)(M1)
+(PHRI)(U1,UMGB)(U1)(11;-0.5)(B3011)(M1)
+(PHRI)(U1,UMGB)(U2)(12;-0.5)(B3011)(M1)
+(PHRI)(U2,UMGB)(U1)(21;-0.5)(B3011)(M1)
+(PHRI)(U2,UMGB)(U2)(22;-0.5)(B3011)(M1)
+(PHRI)(U1,UMGB)(U1)(13;-0.5)(B3012)(M1)
+(PHRI)(U1,UMGB)(U2)(14;-0.5)(B3012)(M1)
+(PHRI)(U2,UMGB)(U1)(32;-0.5)(B3012)(M1)
+(PHRI)(U2,UMGB)(U2)(24;-0.5)(B3012)(M1)
+(PHRI)(U1,UMGB)(U1)(31;-0.5)(B3021)(M1)
+(PHRI)(U1,UMGB)(U2)(32;-0.5)(B3021)(M1)
+(PHRI)(U2,UMGB)(U1)(41;-0.5)(B3021)(M1)
+(PHRI)(U2,UMGB)(U2)(42;-0.5)(B3021)(M1)
+(PHRI)(U1,UMGB)(U1)(33;-0.5)(B3022)(M1)
+(PHRI)(U1,UMGB)(U2)(34;-0.5)(B3022)(M1)
+(PHRI)(U2,UMGB)(U1)(43;-0.5)(B3022)(M1)
+(PHRI)(U2,UMGB)(U2)(44;-0.5)(B3022)(M1)
 M1   4  # VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(M1)+()()(U1+U2)(304;0)(B3020)(M1)
+()()(U1+U2)(102;0)(B31P1)(M1)+()()(U1+U2)(304;0)(B32P2)(M1)
+(PHRO)(U1,UMGB)(U1)(11;-0.5)(B3011)(M1)
+(PHRO)(U1,UMGB)(U2)(12;-0.5)(B3011)(M1)
+(PHRO)(U2,UMGB)(U1)(21;-0.5)(B3011)(M1)
+(PHRO)(U2,UMGB)(U2)(22;-0.5)(B3011)(M1)
+(PHRO)(U1,UMGB)(U1)(13;-0.5)(B3012)(M1)
+(PHRO)(U1,UMGB)(U2)(14;-0.5)(B3012)(M1)
+(PHRO)(U2,UMGB)(U1)(32;-0.5)(B3012)(M1)
+(PHRO)(U2,UMGB)(U2)(24;-0.5)(B3012)(M1)
+(PHRO)(U1,UMGB)(U1)(31;-0.5)(B3021)(M1)
+(PHRO)(U1,UMGB)(U2)(32;-0.5)(B3021)(M1)
+(PHRO)(U2,UMGB)(U1)(41;-0.5)(B3021)(M1)
+(PHRO)(U2,UMGB)(U2)(42;-0.5)(B3021)(M1)
+(PHRO)(U1,UMGB)(U1)(33;-0.5)(B3022)(M1)
+(PHRO)(U1,UMGB)(U2)(34;-0.5)(B3022)(M1)
+(PHRO)(U2,UMGB)(U1)(43;-0.5)(B3022)(M1)
+(PHRO)(U2,UMGB)(U2)(44;-0.5)(B3022)(M1)
```

Figure 5.4 *AKCESS*.AERO *REMI* template for $\{FM1\}_e$

As stated previously, for $\alpha=0=\gamma$ in the TWS^h procedure, (5.4) is universal for all $\{Q\}$ in the form

$$[M]_e\left\{Q^p_{n+1}-Q_n\right\}_e=[\mathrm{M200E}]\left\{Q^p_{n+1}-Q_n\right\}_e \tag{5.44}$$

All surface integrals created in the residual (5.5) by the divergence theorem are handled as discussed in the previous section. Therefore, the following residual expressions contain only terms on the generic FE domain Ω_e. By state variable member:

$$\{\mathrm{RR}\}_e=-[M2J0]_e\{MJ\}_e+\beta_R\hat{\mathrm{U}}K_e\{\mathrm{U}J\}^T_e[\mathrm{M30}KJ]_e\{\mathrm{RHO}\}_e \tag{5.45}$$

$$\begin{aligned}\{\mathrm{RE}\}_e=&-\{\mathrm{U}J\}^T_e[\mathrm{M30}J0]_e\{\mathrm{ETOT}-(\mathrm{Eu}/\mathrm{Ec})\mathrm{PRES}\}_e\\&+\ \mathrm{Pe}^{-1}[\mathrm{M2}KK]_e\{\mathrm{TEMP}\}+\beta_e\hat{\mathrm{U}}K\{\mathrm{U}J\}^T_e[\mathrm{M30}KJ]_e\{\mathrm{ETOT}\}_e\end{aligned} \tag{5.46}$$

$$\begin{aligned}\{\mathrm{RM1}\}_e=&-\{\mathrm{U}J\}^T_e[\mathrm{M30}J0]_e\{\mathrm{M1}\}_e+\mathrm{Eu}[\mathrm{M201}]_e\{\mathrm{PRES}\}_e\\&+\mathrm{Re}^{-1}[\mathrm{M2}KK]_e\{\mathrm{U1}\}_e+\mathrm{Re}^{-1}[\mathrm{M21}K]_e\{\mathrm{U}K/3\}_e\\&+\beta_1\hat{\mathrm{U}}K_e\{\mathrm{U}J\}^T_e[\mathrm{M30}KJ]_e\{M1\}_e\end{aligned} \tag{5.47}$$

$$\begin{aligned}\{\mathrm{RM2}\}_e=&-\{\mathrm{U}J\}^T_e[M30J0]_e\{\mathrm{M2}\}_e+\mathrm{Eu}[\mathrm{M202}]_e\{\mathrm{PRES}\}_e\\&+\mathrm{Re}^{-1}[\mathrm{M2}KK]_e\{\mathrm{U2}\}_e+\mathrm{Re}^{-1}[\mathrm{M22}K]_e\{\mathrm{U}K/3\}_e\\&+\beta_2\hat{\mathrm{U}}K_e\{\mathrm{U}J\}^T_e[\mathrm{M30}KJ]_e\{M2\}_e\end{aligned} \tag{5.48}$$

$$\begin{aligned}\{\mathrm{RM3}\}_e=&-\{\mathrm{U}J\}^T_e[\mathrm{M30}J0]_e\{\mathrm{M3}\}_e+\mathrm{Eu}[\mathrm{M203}]_e\{\mathrm{PRES}\}_e\\&+\mathrm{Re}^{-1}[\mathrm{M2}KK]_e\{\mathrm{U3}\}_e+\mathrm{Re}^{-1}[\mathrm{M23}K]_e\{\mathrm{U}K/3\}_e\\&+\beta_3\hat{\mathrm{U}}K_e\{\mathrm{U}J\}^T_e[\mathrm{M30}KJ]_e\{M3\}_e\end{aligned} \tag{5.49}$$

The next step is to derive the (Newton) jacobian (5.11), which is constructed as the assembly of

$$[JAC]_e\equiv\frac{\partial\{FQ\}_e}{\partial\{Q\}_e}=[M]_e+\theta\Delta t\frac{\partial\{RQ\}_e}{\partial\{Q\}_e} \tag{5.12}$$

The $[M]_e$ term is common to all residuals, however the derivatives of $\{Q\}_e$ requires use of the chain rule. In general, column matrix differentiation by a column matrix is the expression preceding the post-multiplication column matrix. For example, for $\{\mathrm{RR}\}_e$ using (5.45),

$$\frac{\partial\{RR\}_e}{\partial\{RHO\}_e}=\frac{\partial}{\partial\{RHO\}_e}\left(\theta\Delta t\beta_R\hat{U}K_e\{UJ\}_e^T[M30KJ]_e\{RHO\}_e\right)=\theta\Delta t\beta_R\hat{U}K_e\{UJ\}_e^T[M30KJ]_e$$

$$+\theta\Delta t\beta_R\hat{U}K_e\{RHO\}_e^T[M3JK0]_e\frac{\partial\{MJ/RHO\}_e}{\partial\{RHO\}_e}+\theta\Delta t\beta_R UJ_e\{RHO\}_e^T[M3KJ0]_e\frac{\partial\{\hat{M}K/RHO\}_e}{\partial\{RHO\}_e} \tag{5.50}$$

where juxtaposition of pre- and post-multiplication column matrices coincides with tensor index rearrangement in the element hypermatrix [M3...]. Note also that after $\{UJ\}_e$ and $U\hat{K}_e$ transportation, these data are replaced by their nodal state variable definitions and the element-average definition is interchanged such that a matrix differentiation always results.

The matrix derivatives appearing in the last two rows of (5.50) are readily formed using calculus rules, *e.g.*,

$$\frac{\partial\{MJ/RHO\}_e}{\partial\{RHO\}_e}=-\lceil MJ/(RHO)^2\rfloor\!=\!\lceil UJ/(RHO)^2\rfloor \tag{5.51}$$

and $|UJ/RHO|_e$ is a *diagonal* element-order matrix with entries equal to nodal velocity divided by density. Hence, substituting (5.51) into (5.50), and recognizing that (K,J) are "dummy" indices, yields

$$\frac{\partial\{RR\}_e}{\partial\{RHO\}_e}\equiv[RR,R]_e=\theta\Delta t\beta_R\hat{U}K_e\{UJ\}_e^T[M30KJ]_e \tag{5.52}$$

$$-2\theta\Delta t\beta_R\hat{U}K_e\{RHO\}_e^T[M3JK0]_e\lceil UJ/RHO\rfloor_e$$

The "off-diagonal" jacobian matrix $[RR,E]_e$, i.e., the derivative of $\{RR\}_e$ with respect to $\{ETOT\}_e$, is empty, since $\{ETOT\}_e$ is not present explicitly or implicitly in $\{RR\}_e$. The jacobian matrix $[RR,MI]$ has several terms, as follows from (5.45)

$$\frac{\partial\{RR\}_e}{\partial\{MI\}_e}=\theta\Delta t\frac{\partial}{\partial\{MI\}_e}\left([M2J0]_e\{MJ\}_e+\beta_R\hat{U}K_e\{RHO\}_e^T[M3KJ0]_e\{UJ\}_e\right)$$

$$=\theta\Delta t[M2I0]_e+\theta\Delta t\beta_R\hat{U}K_e\{RHO\}_e^T[M3KJ0]_e\frac{\partial\{MJ/RHO\}_e}{\partial\{MI\}_e} \tag{5.53}$$

$$+\theta\Delta t\beta_R UJ_e\{RHO\}_e^T[M3JK0]_e\frac{\partial\{\hat{M}K/RHO\}_e}{\partial\{MI\}_e}$$

The derivative of $\{MJ\}_e$ with respect to $\{MI\}_e$ is the Kronecker delta, as shown in the $[M2I0]_e$ term, hence the second/third right-side terms in (5.53) involve the diagonal matrix

$$\frac{\partial\{MJ/RHO\}_e}{\partial\{MI\}_e} = \delta_{IJ}\lceil 1/RHO \rfloor \tag{5.54}$$

Labeling inverse density as "OSRH," i.e., one-slash-rho, (5.53 becomes

$$[RR,MI]_e = \theta\Delta t[M2I0]_e + 2\theta\Delta t\beta_R\hat{U}K_e\{RHO\}_e^T[M3KJ0]_e\lceil OSRH\rfloor_e \tag{5.55}$$

Note that "I" is a free tensor index in (5.55), hence there are d expressions contained therein.

Equations (5.52) and (5.55) express the element matrices filling the first row (by blocks) of $[JAC]_e$, (5.12). The second (block) row therein contains the Newton jacobian contributions for total energy residual $\{RE\}_e$, which brings the requirement to handle the equation of state variable form (5.7). The self-differentiated jacobian is

$$\begin{aligned}[RE,E]_e = &-\theta\Delta t\{UJ\}_e^T[M30J0]_e + \theta\Delta t EuEc^{-1}\{UJ\}_e^T[M30J0]_e\frac{\partial\{PRES\}_e}{\partial\{ETOT\}_e} \\ &+\theta\Delta t Pe^{-1}[M2KK]_e\frac{\partial\{TEMP\}_e}{\partial\{ETOT\}_e} + \theta\Delta t\beta_E\hat{U}K_e\{UJ\}_e^T[M30JK]_e\end{aligned} \tag{5.56}$$

Equation (5.7) contains the equation of state conservation law form, hence

$$\frac{\partial\{PRES\}_e}{\partial\{ETOT\}_e} = \frac{\partial}{\partial\{ETOT\}_e}\left(\frac{(\gamma-1)Ec}{Eu}\right)\{ETOT\}_e = \frac{(\gamma-1)Ec}{Eu}\lceil I\rfloor \tag{5.57}$$

where $\lceil I\rfloor$ is the identify matrix. Using (5.57) and (2.6), via the chain rule

$$\frac{\partial\{TEMP\}_e}{\partial\{ETOT\}_e} = \frac{\partial\{TEMP\}_e}{\partial\{PRES\}_e}\frac{\partial\{PRES\}_E}{\partial\{ETOT\}_e} = \lceil OSRH\rfloor_e\frac{(\gamma-1)Ec}{Eu}\lceil I\rfloor \tag{5.58}$$

for the non-dimensionalization of (2.6) via a uniform reference thermodynamic state, recall Section 2.4. If p_r is selected otherwise, then a non-D gas constant will appear in (5.58). With (5.50)-(5.58), (5.56) becomes

$$\begin{aligned}[RE,E]_e = &-\theta\Delta t\{UJ\}_e^T[M30J0]_e + \theta\Delta t\frac{(\gamma-1)mk}{Ec}\{UJ\}_e^T[M30J0]_e \\ &+\theta\Delta t\frac{(\gamma-1)Ec}{PeEu}[M2KK]_e\lceil OSRH\rfloor_e + \theta\Delta t\beta_E\hat{U}K_e\{UJ\}_e^T[M30KJ]_e\end{aligned} \tag{5.59}$$

The off-diagonal jacobians $[RE,R]_e$ and $[RE,MI]_e$ involve chain-rule operations for appearance of pressure and velocity in the convection and dissipation terms or$[]_e$

$$[\mathrm{RE,R}]_e = -\{\mathrm{ETOT} + (\mathrm{Ec}/\mathrm{Eu})\mathrm{PRES}\}_e^T [\mathrm{M30}J0]_e \frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{RHO}\}_e}$$

$$\frac{\mathrm{Ec}}{\mathrm{Eu}}\{\mathrm{U}J\}_e^T [\mathrm{M30}J0]_e \frac{\partial\{\mathrm{PRES}\}_e}{\partial\{(\mathrm{RHO}\}_e} \tag{5.60}$$

$$+\frac{1}{\mathrm{Pe}}[\mathrm{M2}KK]\frac{\partial\{\mathrm{TEMP}\}_e}{\partial\{\mathrm{RHO}\}_e} + 2\beta_E \hat{\mathrm{U}}K_e\{\mathrm{ETOT}\}_e^T [\mathrm{M3}JK0]_e \frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{RHO}\}_e}$$

All but the second and third term derivatives are established. For the later

$$\frac{\partial\{\mathrm{TEMP}\}_e}{\partial\{\mathrm{RHO}\}_e} = \frac{\partial\{\mathrm{PRES}/\mathrm{RHO}\}_e}{\partial\{\mathrm{RHO}\}_e} = -\lceil \mathrm{PRES}/(\mathrm{RHO})^2 \rfloor_e = -\lceil \mathrm{TEMP}/\mathrm{RHO} \rfloor_e \tag{5.61}$$

for the consistent non-dimensionalization. The second derivative can be expressed via (5.7), the state variable form, or the equation of state (2.6). For the former, and element-averaging one momentum term

$$\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{RHO}\}_e} = \frac{\partial}{\partial\{\mathrm{RHO}\}_e}\left(\mathrm{Ec}\{\mathrm{ETOT}\}_e - \frac{1}{2}\mathrm{M}K\{\mathrm{M}K/\mathrm{RHO}\}_e\right)$$

$$= +\frac{(\gamma-1)}{2\mathrm{Eu}}\mathrm{M}K_e \lceil \mathrm{M}K/(\mathrm{RHO})^2 \rfloor_e = \frac{(\gamma-1)}{2\mathrm{Eu}} \lceil \mathrm{U}K\mathrm{U}K \rfloor_e \tag{5.62}$$

Conversely, using (2.6) with consistent non-D

$$\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{RHO}\}_e} = \lceil \mathrm{TEMP} \rfloor_e \tag{5.63}$$

Hence, replacing the diagonal matrix argument in (5.61) with $\lceil \mathrm{TSRH} \rfloor$, i.e., "$T$ slash ρ," and using (5.51), (5.61)-(5.63), (5.60) becomes

$$[\mathrm{RE,R}]_e = \theta\Delta t\{\mathrm{ETOT} + (\mathrm{Ec}/\mathrm{Eu})\mathrm{PRES}\}_e^T [\mathrm{M30}J0]_e \lceil \mathrm{U}J\mathrm{SR} \rfloor_e$$

$$+\frac{\theta\Delta t(\gamma-1)\mathrm{Ec}}{2\mathrm{Eu}^2}\{\mathrm{U}J\}_e^T [\mathrm{M30}J0]_e \lceil \mathrm{U}K\mathrm{U}K \rfloor_e$$

$$-\frac{\theta\Delta t}{\mathrm{Pe}}[\mathrm{M2}KK]_e \lceil \mathrm{TSRH} \rfloor_e$$

$$-2\theta\Delta t\beta_E \hat{\mathrm{U}}K_e\{\mathrm{ETOT}\}_e^T [\mathrm{M3}JK0]_e \lceil \mathrm{U}J\mathrm{SR} \rfloor_e \tag{5.64}$$

The derivation of the jacobian $[\mathrm{RE,M}J]_e$ involves similar details; using (5.46)

$$[\mathrm{RE},\mathrm{M}I]_e = -\theta\Delta t\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]\frac{\mathrm{Eu}}{\mathrm{Ec}}\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{M}I\}_e}$$

$$-\theta\Delta t\{\mathrm{ETOT}+(\mathrm{Eu}/\mathrm{Ec})\mathrm{PRES}\}_e^T[\mathrm{M30}J0]\frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{M}I\}_e} \tag{5.65}$$

$$+\frac{\theta\Delta t}{\mathrm{Pe}}[\mathrm{M2}KK]\frac{\partial\{\mathrm{TEMP}\}_e}{\partial\{\mathrm{M}I\}_e}+2\theta\Delta t\hat{\mathrm{U}}K\{\mathrm{ETOT}\}_e^T[\mathrm{M3}KJ0]\frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{M}I\}_e}$$

By terms, the new chain rule derivatives needed are

$$\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{M}I\}_e}=\frac{-(\gamma-1)}{2\mathrm{Eu}}\frac{\partial}{\partial\{\mathrm{M}I\}_e}\left\{\mathrm{RHO}\frac{\mathrm{M}K^2}{}\right\}_e=-\frac{-(\gamma-1)}{\mathrm{Eu}}\left\lceil\frac{\mathrm{M}K}{\mathrm{RHO}}\right\rfloor\delta_{IK}=-\frac{-(\gamma-1)}{\mathrm{Eu}}\lceil\mathrm{U}I\rfloor_e \tag{5.66}$$

$$\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{MI}\}_e}=\frac{\partial\{\mathrm{PRES}/\mathrm{RHO}\}_e}{\partial(\mathrm{MI})_e}=-\frac{-(\gamma-1)}{2\mathrm{Eu}}\frac{\partial\left\{(\mathrm{MK}/\mathrm{RHO})^2\right\}}{\partial\{\mathrm{MI}\}_e}$$

$$=-\frac{-(\gamma-1)}{\mathrm{Eu}}\lceil\mathrm{M}K/\mathrm{RHO}^2\rfloor\delta_{IK} = -\frac{-(\gamma-1)}{\mathrm{Eu}}\lceil\mathrm{U}I\mathrm{SR}\rfloor_e \tag{5.67}$$

hence (5.65) becomes

$$[\mathrm{RE},\mathrm{M}I]_e=\frac{\theta\Delta t(\gamma-1)}{\mathrm{Ec}}\left\{\mathrm{U}J_e^T[\mathrm{M30}J0]_e\right\}\lceil\mathrm{U}I\rceil_e$$

$$-\theta\Delta t\{\mathrm{ETOT}+(\mathrm{Eu}/\mathrm{Ec})PRES\}_e^T[\mathrm{M30}I0]_e\lceil\mathrm{OSRH}\rfloor_e$$

$$\frac{-\theta\Delta t(\gamma-1)}{\mathrm{PeEu}}[\mathrm{M2}KK]_e\lceil\mathrm{U}I\mathrm{SR}\rfloor_e+2\theta\Delta t\hat{\mathrm{U}}K_e\{\mathrm{ETOT}\}_e^T[\mathrm{M3}KI0]_e\lceil\mathrm{OSRH}\rfloor_e \tag{5.68}$$

and this completes the Newton jacobian for the energy variable.

The same matrix calculus operations lead to the momentum TWS^h statement jacobian. The coupling to density yields, using (5.47)-(5.49) with I the free index

$$[\mathrm{RM}I,\mathrm{R}]_e=-\theta\Delta t\{\mathrm{M}I\}_e^T[\mathrm{M30}J0]_e\frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{RHO}\}_e}$$

$$+\theta\Delta t\mathrm{Eu}[\mathrm{M20}I]_e\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{RHO}\}_e}+\frac{\theta\Delta t}{\mathrm{Re}}[\mathrm{M2}KK]\frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{RHO}\}_e} \tag{5.69}$$

$$+\frac{\theta\Delta t}{\mathrm{Re}}[\mathrm{M2}IK]_e\frac{\partial\{\mathrm{M}K/\mathrm{RHO}\}}{\partial\{\mathrm{RHO}\}_e}+2\theta\Delta t\beta_I\hat{\mathrm{U}}K_e\{\mathrm{MI}\}_e^T[\mathrm{M3}TK0]_e\frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{RHO}\}_e}$$

All needed chain rule derivatives are available, hence the final form is

$$
\begin{aligned}
[\mathrm{RM}I,\mathrm{R}]_e = &+\theta\Delta t\{\mathrm{M}I\}_e^T[\mathrm{M30}J0]_e\lceil \mathrm{U}J\mathrm{SR}\rfloor_e + \frac{\theta\Delta t(\gamma-1)}{2}[\mathrm{M20}I]_e\lceil \mathrm{UKUK}\rfloor_e \\
&-\frac{\theta\Delta t}{\mathrm{Re}}[\mathrm{M2}KK]_e\ \lceil \mathrm{U}I\mathrm{SR}\rfloor_e - \frac{\theta\Delta t}{\mathrm{Re}}[\mathrm{M2}IK]_e\ \lceil \mathrm{UKSR}\rfloor_e \\
&-2\theta t\beta_I\hat{\mathrm{U}}K_e\{\mathrm{M}I\}_e^T[\mathrm{M3}JK0]_e\ \lceil \mathrm{U}J\mathrm{SR}\rfloor_e
\end{aligned}
\tag{5.70}
$$

The derivative of $\{\mathrm{FM}I\}_e$ with respect to the energy variable directly produces the jacobian via (5.7) as

$$
[\mathrm{RM}I,\mathrm{E}]_e = \mathrm{Eu}[\mathrm{M20}I]_e\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{ETOT}\}_e} = (\gamma-1)\mathrm{Ec}[\mathrm{M20}I]_e \tag{5.71}
$$

The self-coupling jacobian contributions are

$$
\begin{aligned}
&\frac{\partial\{\mathrm{RM}I\}_e}{\partial\{\mathrm{M}L\}_e}\delta_{IL} \equiv [\mathrm{RM}\underline{I},\mathrm{M}\underline{I}]_e \\
&= \theta\Delta\Bigg(\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]_e\frac{\partial\{\mathrm{M}I\}}{\partial\{\mathrm{ML}\}} - \{\mathrm{M}I\}_e^T[\mathrm{M30}J0]_e\frac{\partial\{\mathrm{M}J/\mathrm{RHO}\}_e}{\partial\{\mathrm{M}L\}_e} \\
&+\mathrm{Eu}[\mathrm{M20}I]_e\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{M}L\}_e} + \frac{1}{\mathrm{Re}}[\mathrm{M2}KK]_e\frac{\partial\{\mathrm{M}I/\mathrm{RHO}\}_e}{\partial\{\mathrm{M}L\}_e} \\
&+\frac{1}{3\mathrm{Re}}[\mathrm{M2}IK]_e\frac{\partial\{\mathrm{M}K/\mathrm{RH})\}_e}{\partial\{\mathrm{M}L\}_e} + \beta_I\hat{\mathrm{U}}K_e\{\mathrm{UJ}\}_e^T[\mathrm{M30}KJ]_e\frac{\partial\{\mathrm{M}I\}_e}{\partial\{\mathrm{M}L\}_e} \\
&+2\beta_I\hat{\mathrm{U}}K_e\{\mathrm{M}I\}_e^T[\mathrm{M3}JK0]\frac{\partial\{\mathrm{MJ}/\mathrm{RHO}\}_e}{\partial\{\mathrm{ML}\}_e}\Bigg)\delta_{IL}
\end{aligned}
\tag{5.72}
$$

Enforcing the Kronecker delta, using (5.66), and for subscript bar denoting *not* a summation index,

$$
\begin{aligned}
[\mathrm{RM}\underline{I}\mathrm{M}\underline{I}]_e = &-\theta\Delta t\{\mathrm{U}J\}_e^T[\mathrm{M30}J0] - \theta\Delta t\{\mathrm{M30}I0\}_e\lceil \mathrm{OSRH}\rfloor_e \\
&-\theta\Delta t(\gamma-1)[\mathrm{M20}I]_e\lceil \mathrm{U}\underline{I}\rfloor_e + \frac{\theta\Delta t}{\mathrm{Re}}[\mathrm{M3}KK]_e\lceil \mathrm{OSRH}\rfloor_e \\
&+\frac{\theta\Delta t}{3\mathrm{Re}}[\mathrm{M3}\underline{II}]_e\lceil \mathrm{OSRH}\rfloor_e + \theta\Delta t\beta_I\hat{\mathrm{U}}K_e\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e \\
&+\theta\Delta t\beta_I\hat{\mathrm{U}}K_e\{\mathrm{M3}IK\}_e\lceil \mathrm{OSHR}\rfloor
\end{aligned}
\tag{5.73}
$$

The non-self coupled jacobians are also derived from (5.72), by removing the Kronecker delta and replacing it with the constraint $I \neq L$. Then, the remaining contributions to the TWS^h algorithm Newton jacobian include

$$
\begin{aligned}
[\mathrm{RM}I,\mathrm{M}J]_e = & -\theta\Delta t\{\mathrm{M}I\}_e^T[\mathrm{M30}J0]_e \lceil \mathrm{OSRH} \rfloor_e \\
& -\theta\Delta t(\gamma-1)[\mathrm{M20}I]_e \lceil \mathrm{UJ} \rfloor_e + \frac{\theta\Delta t}{3\mathrm{Re}}[\mathrm{M2}IJ]_e \lceil \mathrm{OSRH} \rfloor_e \\
& +2\theta\Delta t\beta_I \hat{\mathrm{U}}K_e\{\mathrm{M}I\}_e^T[\mathrm{M3}JK0]_e \lceil \mathrm{OSRH} \rfloor_e
\end{aligned}
\tag{5.74}
$$

5.5 Summary

This section has developed the complete matrix statement of the FE TWS^h CFD algorithm for compressible aerodynamic flow prediction. The "computable" form is the Newton algorithm (3.17), each contribution to which is constructed as the *assembly* of (5.4),(5.5) and (5.12), over the elements Ω^e of the discretization Ω^h. Gauss quadrature is available to construct each associated integral to definable precision, and a significant reduction in computer execution cost accrues to an averaged-metric formulation that yields universal element *master matrices*. Implementation of boundary conditions, as well as both derived TWS^h dissipation mechanisms, is detailed. For reference, Appendix B collates the TWS^h algorithm matrix statements, and Appendix C contains the *AKCESS.* template* for d=2.

6. AUXILIARY PROCEDURES, LINEAR ALGEBRA

6.1 Synopsis

The FE TWS^h CFD algorithm for aerodynamics applications is defined, including imposing of admissible boundary conditions and two numerical dissipation algorithms. Auxiliary procedures complementing this algorithm include initial condition generation, an implicit Runge-Kutta ODE algorithm, an equilibrium real-air solution procedure and a matrix tensor product factorization of the Newton jacobian yielding a block-($2k$+1) matrix quasi-Newton procedure. Each of these options exhibits a certain computational efficiency potential and this section derives and presents these auxiliary methodologies.

6.2 Initial condition generation

The RaNS/E conservation law system is tightly coupled in its state variable $q(\mathbf{x},t)$, in concert with the equation of state $p=p(q)$, as an initial-boundary value problem. The boundary condition (BC) well-posedness issue is well resolved, however this is not so for an initial condition (IC) specification. This IC problem is exacerbated when the freestream flow is subsonic, since to specify the principal momentum component is not an admissible BC.

One "industry-standard" resolution is to uni-directionally impose the design freestream principal momentum component everywhere, then let the code "crunch" long enough to self-generate an Euler, BC-satisfying state variable distribution $\{Q(t>t_0)\}$. An alternative is interpolation of quasi-one dimensional analytical solutions (isentropic, inviscid) onto the geometry such that flow tangency is at least an IC property, cf., Iannelli (1991). With the additional assumption of irrotational, a tangency-and thermodynamics-true, multi-dimensional IC generation procedure accrues to a compressible potential flow solution.

Following experimentation with the alternatives, an iterative potential flow IC generation process that includes both tangency and thermodynamics is developed. It constitutes a pair of Galerkin weak statements, along with the isentropic/adiabatic stagnation energy analytical solution. The non-D steady state form for continuity is $\nabla\cdot\mathbf{m}=0$, which coupled with the irrotational assumption $\mathbf{u}=-\nabla\phi$ yields

$$L(\phi)=-\nabla\cdot\rho\nabla\phi=0 \tag{6.1}$$

On flow tangency boundaries, $\nabla\phi\cdot\hat{n}=0$, while at inflow and/or outflow boundaries

$$-\nabla\phi\cdot\hat{\mathbf{n}}=\begin{cases}-U_{in}\\ +U_{out}\end{cases} \tag{6.2}$$

where U_α is assumed given data. Recalling Table 2.1, for subsonic inflow, specification of U_{in} is inadmissible; however, $\mathbf{m}\cdot\hat{\mathbf{s}}=0$ is appropriate, hence $\phi(\mathbf{x}_{in})=0$ is the available Dirichlet constraint.

The Galerkin weak statement for (6.1)-(6.2) is

$$WS^N \equiv \int_\Omega \{N\}L(\phi)d\tau=0 \tag{6.3}$$

which when discretized, and in the developed nomenclature, yields the computable residual $\{R\phi\}=\underset{e}{S}\{R\phi\}_e$ and

$$\{R\phi\}_e=\{\mathrm{RHO}\}_e^T[\mathrm{M30}KK]_e\{\mathrm{PHI}\}_e+\{\mathrm{RHO}\}_e^T[\mathrm{S3000}]_e\{UJNJ\}_e=[0] \tag{6.4}$$

In (6.4), $\{\mathrm{RHO}\}_e$ is the nodal distribution of density on Ω_e, while $\{UJNJ\}_e$ is the surface nodal efflux vector implementation of (6.2).

Density is an unknown in (6.4), hence an iterative strategy is required for its estimation. Therefore, solution for (6.4) is cast in Newton algorithm form (3.17), with element-level quasi-Newton jacobian

$$[JAC]_e\equiv[R\phi,\phi]_e=\{\mathrm{RHO}\}_e^T[\mathrm{M30}KK]_e \tag{6.5}$$

where (6.4), at iteration level p, is the equivalent of $\{F\phi\}_e^p$.

The output from (6.4)-(6.5) is the pth estimate of discrete potential function, from which is required to extract a velocity distribution estimation for the adiabatic energy equation

$$h=h_0-\frac{1}{2}u^2 \tag{6.6}$$

where subscript "zero" is the stagnation reference state for enthalpy h. For arbitrary meshing, a Galerkin weak statement on the defining equation for potential function is

$$WS^N(\mathbf{u})=\int_\Omega \{N\}(\mathbf{u}+\nabla\phi)^N d\tau\equiv 0 \tag{6.7}$$

The discretization of (6.7), in element level notation, produces the $1<I<d$ algebraic equations

$$\{\mathrm{RU}I\}_e=[\mathrm{M200}]_e\{\mathrm{U}I\}_e^T+[\mathrm{M20}I]_e\{\mathrm{PHI}\}_e^p=\{0\} \tag{6.8}$$

which are linear and separable. No BC requirements exist, since the divergence theorem was not used, and (6.8) is directly solvable in global form as

$$[\mathrm{M200}]\{\mathrm{U}I\}=-[\mathrm{M20}I]\{\mathrm{PHI}\} \tag{6.9}$$

The output from (6.9) is the pth estimation of the nodal velocity field. For reference, Fig. 6.1 shows the two group *AKCESS.* template* for (6.4)-(6.5) and (6.8)-(6.9). With $\{UI\}_e^p$, (6.6) in non-D variables yields at every node of Ω^h

$$T = T_0 - \frac{(\gamma-1)}{2}\mathrm{Ma}_r^2(UIUI) \tag{6.10}$$

where Ma_r is the input reference Mach number, recall (2.20). From the definition of Mach number, and (6.10)

$$\mathrm{Ma} = \mathrm{Ma}_r\sqrt{u^2/T} \tag{6.11}$$

whereupon the isentropic adiabatic stagnation solution for density is

$$\rho = \left(\frac{\rho_0}{\rho_r}\right)\left[1+\frac{(\gamma-1)}{2}\mathrm{Ma}^2\right] \tag{6.12}$$

where ρ_r is the defined reference state for non-D, which may differ from stagnation density ρ_0.

Equation (6.12) provides the pth estimate for density (distribution on Ω^h), which is then inserted into (6.4) to complete evaluation of $\{R\phi\}^p$. The iterative sequence (6.4)-(6.12) typically converges to $\varepsilon < 10^{-4}$ in 3-4 iterations for subsonic free stream flows with only a local supersonic bubble. At convergence the (non-D) pressure distribution is completed as

$$p = \frac{p_0}{p_r}\left[1+\frac{(\gamma-1)}{2}\mathrm{Ma}^2\right]^{(\gamma-1)/\gamma} \tag{6.13}$$

hence the nodal distributions {UI}, {PRES}, {RHO} and {TEMP} are filled with IC data. It is then an elementary task to establish the corresponding state variable $\{Q(t_0)_e$.

6.3 Implicit Runge-Kutta algorithm

The TWS^h FE algorithm development, (3.13)-(3.17), employed the θ-implicit Euler single step ODE algorithm family for exposition. An alternative *implicit* Runge-Kutta (IRK) ODE algorithm, developed by Iannelli(1991) possesses certain operational features superior to the θ-family for aerodynamics application.

As stated in Section 3, a weak statement for the RaNS/E system always produces an ODE system of the form

$$WS^h = [M]\frac{d\{Q\}}{dt} + \{RQ\} = \{0\} \tag{6.14}$$

The coupling "mass" matrix $[M]$ in (6.14) yields a non-standard ODE form. Hence, conceptually multiply through by $[M]^{-1}$, and then changing to the scalar lower case notation used in the ODE literature, the WS^h-equivalent ODE system is

$$\frac{dq}{dt} = f(q,t) \tag{6.15}$$

In (6.15) form, the θ-implicit Euler ODE algorithm (3.14) is

$$q_{n+1} - q_n = \Delta t\left(\theta f_{n+1} + (1-\theta) f_n\right) \tag{6.16}$$

In comparison, the two-stage, diagonally-implicit Runge-Kutta ODE algorithm is

$$\begin{aligned} q_{n+1} - q_n &= \Delta t\left(b_1 k_1 + b_2 k_2\right) \\ k_1 &= f\left(t_n + c_1 \Delta t, q_n + \alpha_1 \Delta t k_1\right) \\ k_2 &= f\left(t_n + c_2 \Delta t, q_n + \beta_{21} \Delta t k_1 + \alpha_2 \Delta t k_2\right) \end{aligned} \tag{6.17}$$

where subscript n is the time step index and $\Delta t = t_{n+1} - t_n$. For consistency, the IRK coefficients must satisfy constraints $c_1 = \alpha_1$ and $c_2 = \beta_{21} + \alpha_2$, and the usual presentation of (6.17) in the ODE literature is as the *synoptic* table

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} \equiv \begin{array}{c|cc} c_1 & \alpha_1 & 0 \\ c_2 & \beta_{21} & \alpha_2 \\ \hline & b_1 & b_2 \end{array} \tag{6.18}$$

The IRK algorithm coefficient sets contain flexibility for control of truncation error and stability classification. In the classical sense, second-order accuracy accrues to the constraints

$$b_1 + b_2 = 1 \text{ and } b_1\alpha_1 + b_2\left(\beta_{21} + \alpha_2\right) = 0.5 \tag{6.19}$$

Additional desirable properties for the IRK algorithm to possess include the *A-stability* of the θ=1 backwards Euler (which is only first-order accurate) and the classical second-order accuracy of the θ=0.5 trapezoidal rule. For the IRK algorithm to be uniformly second-order accurate for all Δt requires it to possess *B-stability*, the essence of which is expressed as

$$\left|error_{n+1}\right| \le D\Delta t^{p+1}, \text{ for all } \Delta t \in \left(0, \Delta t_1\right) \tag{6.20}$$

where constants D and Δt_1 are independent of ODE system *stiffness*, and p is the order of consistency. The IRK algorithm (6.17) is bounded on $1 < p < 2$, and the sufficient conditions for *B*-consistency of unit order are $\alpha_1 >$ and $\alpha_2 > 0$. Additionally, for the *B*-convergence concept of stiffness-independence of convergence rate to hold, the necessary conditions are $b_1 \le 0$ and $b_2 \le 0$, c.f., Iannelli (1991, Ch. 8). One IRK coefficients satisfying all criteria is

$$\begin{aligned} \alpha_1 &= \frac{3-\sqrt{3}}{6} & \alpha_3 &= 2-\sqrt{3} & b_1 &= \frac{3-\sqrt{3}}{4} \\ \alpha_2 &= \frac{\sqrt{3}-1}{2} & & & b_2 &= \frac{1+\sqrt{3}}{4} \end{aligned} \tag{6.21}$$

For comparison to (6.18), the synoptic table for the θ-implicit Euler ODE algorithm is

$$\begin{array}{c|c|c|cc} & & 0 & 0 & 0 \\ c & A & \equiv 1 & (1-\theta) & \theta \\ \hline & b^T & & (1-\theta) & \theta \end{array}$$

The implementation of the IRK ODE algorithm, as replacement for the θ-implicit Euler algorithm (3.14), hence the Newton algorithm (3.17) involves only minor modifications. The algebraic matrix statement (3.17) becomes two expressions involving the IRK iterates $\{k_\alpha\}^p$ as

$$\begin{aligned} \{FQ\}_1^p &= [M]\{k_1\}^p + \{RQ\}\left(t_n + c_1\Delta t, \{Q_n + a_1\Delta t\{k_1\}^p\right) \\ \{FQ\}_2^p &= [M]\{k_2\}^p + \left\{RQ\left(t_n + c_2\Delta t, \{Q\}_n + \beta_{21}\Delta t\{k_1\}^p + \alpha_2\Delta t\{k_2\}^p\right)\right\} \end{aligned} \tag{6.22}$$

The Newton jacobian (3.17a) similarly becomes the two expressions

$$\begin{aligned} \left[JAC_1\right] &= [M] + \alpha_1\Delta t \left.\frac{\partial\{RQ\}}{\partial\{Q\}}\right|_{\{Q\}_n + \alpha_1\Delta t\{k_1\}^p} \\ \left[JAC_2\right] &= [M] + \alpha_2\Delta t \left.\frac{\partial\{RQ\}}{\partial\{Q\}}\right|_{\{Q\}_n + \beta_{21}\Delta t\{k_1\}^p + \alpha_2\Delta t\{k_2\}^p} \end{aligned} \tag{6.23}$$

The iterates $\{\delta k_1\}^{p+1}$ and $\{\delta k_2\}^{p+1}$ are the solutions to (6.23)-(6.22, hence the time-updated iterate for the discrete state variable at time t_{n+1} is

$$\{Q\}_{n+1}^{p+1} = \{Q\}_n + \Delta t \sum_{i=0}^{p}\left(b_1\{\delta k_1\}^{i+1} + b_2\{\delta k_2\}^{i+1}\right) \tag{6.24}$$

Comparing (3.17) and (6.22)-(6.24) confirms that twice the amount of computational work is required to solve (3.15) using the IRK ODE algorithm. However, verification and benchmark problems reported by Iannelli(1991) indicate that allowable (stable) time steps Δt are also larger for IRK than for backwards Euler (θ=1), and yield a second-order accurate solution process devoid of the numerical diffusion associated with using θ=1.

A *non-iterative* implementation of either algorithm is also possible, enforced by constraining p=0. The first iterate becomes

$$\{\delta(Q)\}^{p+1} \Rightarrow \{\Delta(Q)\} \text{ in (3.17b), or } \{\delta k_i\}^{p+1} \Rightarrow \{\Delta(k_i)\}$$

for i=(1,2) in (6.24), and the IRK procedure (6.22) reduces to the two stage sequence

$$\left[M+\alpha_1\Delta t\frac{\partial\{RQ\}}{\partial\{Q\}}\bigg|_{\{Q\}_n}\right]\{\Delta k_1\}=-\left\{RQ\left(t_n,\{Q\}_n\right)\right\} \tag{6.25}$$

$$\left[M+\alpha_2\Delta t\frac{\partial\{RQ\}}{\partial\{Q\}}\bigg|_{\{Q\}_n^+}\right]\{\Delta k_2\}=-\left\{RQ\left(t_n+c_2\Delta t,\{Q\}_n^+\right)\right\} \tag{6.26}$$

where $\{Q\}_n^+\equiv\{Q\}_n+\beta_{21}\Delta t\{\Delta k_1\}$ is the intermediate state variable. The summation in (6.24) vanishes, hence

$$\{Q\}_{n+1}=\{Q\}_n+\Delta t\left(b_1\{\Delta k_1\}+b_2\{\Delta k_2\}\right) \tag{6.27}$$

is the time-advanced discrete state variable.

6.4 Equilibrium reacting air algorithm

As introduced in Section 2, long-term aerodynamics interest resides in inclusion of real-gas effects, as occurs for example at hypersonic Mach number. The five species generalization of the internal energy definition is (2.9), and (2.11)-(2.13) are five algebraic constraint equations on species mass fraction Y_α. Hence, (2.8)-(2.13) represent seven equations on the seven variables p, T and Y_α. The state variable solution to (2.1)-(2.3) yields ρ, E and m_i, hence $u^2 = m_i m_i/\rho^2$, and mass-specific internal becomes available via $(E+p/\rho)-u^2/2=c_pT$. Hence, from (2.9)

$$c_pT=\sum_{i=1}^{5}Y_ic_{pi}T+\sum_{i=3}^{5}Y_i\frac{R\theta_i^v/M_i}{\exp\left(\theta_i^v/T\right)-1}+\sum_{i=1}^{3}Y_ih_i^0 \tag{6.28}$$

which decouples pressure from the solution strategy.

Next, since (2.12)-(2.13) are linear, they can be rearranged to the explicit expressions

$$\begin{aligned}Y_1-\left(\alpha_{10}-Y_4+\alpha_{13}Y_3\right)&=0\\ Y_2-\left(\alpha_{20}-Y_5+\alpha_{23}Y_3\right)&=0\end{aligned} \tag{6.29}$$

where the α_{ij} are functions only of the respective molecular masses. Insertion of (6.29) into (2.11) then produces the companion expressions

$$Y_5 = \left(\alpha_{20} + \alpha_{23} Y_3 + \frac{M_2^2 K_1}{2\rho M_5}\right) - \left(\frac{M_2^2 K_1}{2\rho M_5}\left(\alpha_{20} + \alpha_{23} Y_3 + \frac{M_2^2 K_1}{4\rho M_5}\right)\right)$$

$$Y_4 = \left(\alpha_{10} + \alpha_{13} Y_3 + \frac{M_1^2 K_2}{2\rho M_4}\right) - \left(\frac{M_1^2 K_2}{\rho M_4}\left(\alpha_{10} + \alpha_{13} Y_3 + \frac{M_1^2 K_2}{4\rho M_4}\right)\right) \tag{6.30}$$

Hence, (6.29)-(6.30) depend only on Y_3 and T, which yields a significant reduction in work for a solution.

The final algebraic equation pair for solution is (Iannelli, 1991)

$$f_1 \equiv Y_3 - M_3\left(Y_4 Y_5 \frac{K_3}{M_4 M_5}\right)^{1/2} = 0$$

$$f_1 \equiv e - RT\sum_{i=1}^{5}\frac{Y_i}{M_i} + \sum_{i=1}^{5} Y c_{pi} T + \sum_{i=1}^{3} Y h_i^0 - \sum_{i=3}^{5} Y_i \frac{R\theta_i^v / M_i}{\exp\left(\theta_i^v / T\right) - 1} - u^2 = 0 \tag{6.31}$$

where $e = E/\rho$ and $u = m_i m_i / \rho^2$. Hence, (6.31) is directly related to the discrete state variable solution $q^h \Rightarrow \{Q(t_n)\}$, and to Y_α, α=1,2,4,5 via (6.29)-(6.30).

The solution of (6.31) is cast as the Newton algorithm

$$\{F(Y,T)\} = \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = \{0\} \tag{6.32}$$

$$\begin{Bmatrix} \delta Y \\ \delta T \end{Bmatrix}^{p+1} = -\left[\frac{\partial\{F\}}{\partial\{Y,T\}}\right]^{-1} \{F\}^p \tag{6.33}$$

The initial condition for starting (6.32)-(6.33) is the previous nodal solution (at time t_n), or the solution at the node adjacent to the current iteration. Since the jacobian in (6.33) is only 2x2, an analytical expression for the inverse can be derived.

6.5 Tensor matrix product factorization

As mentioned, the FE TWS^h algorithm (3.17) is a non-linear algebraic equation system, for which the Newton jacobian (3.17a) is typically a large matrix for the choice of an implicit time-integration procedure. A major focus in this project was to develop, test and validate a tensor matrix product factorization (TP) quasi-Newton replacement for implicit efficiency. The matrix factorization concept has roots in ADI and approximate factorization (AF) finite difference methods, c.f., Beam and Warming(1976).

The specific TP factorization for a FE WS^h relies mathematically on existence of the non-diagonal "mass matrix" $[M]$ in the Newton jacobian. Referring to Baker & Pepper(1991, Ch. 8) for details, and realizing that any (global) matrix is always the

assembly of element-level contributions, then (5.12) is the basic departure point for a linear algebra approximation, i.e.,

$$[\mathrm{JAC}]_e = [\mathrm{M}]_e + \theta\Delta t \frac{\partial\{RQ\}_e}{\partial\{Q\}_e} \tag{6.34}$$

Further realizing that the residual derivative contributions in (6.34) have a directional character, via the flux vector resolutions, then a *generic* form for (6.34) is

$$\begin{aligned}[\mathrm{JAC}]_e = [\mathrm{M}]_e + \theta\Delta t\big([\mathrm{V1}]_e + [\mathrm{V2}]_e + [\mathrm{V3}]_e + [\mathrm{D1}]_e \\ + [\mathrm{D2}]_e + [\mathrm{D3}]_e + [\mathrm{DB1}]_e + \cdots\big)\end{aligned} \tag{6.35}$$

In (6.35), $[\mathrm{V}I]_e$ $[\mathrm{D}I]_e$ and $[\mathrm{DB}I]_e$ for $1 < I < d$ are the η-space resolutions of the general expressions for the convection, diffusion and β-term artificial diffusion matrix contributions, as discussed (and derived) in detail in Section 5.

For exposition, assume that the x_i and η_j coordinate systems are parallel, *i.e.*, Ω^h is rectangular cartesian. Then, the TP factorization of (6.35) is

$$[\mathrm{JAC}I]_e \equiv [\mathrm{JAC1}]_e \otimes [\mathrm{JAC2}]_e \otimes [\mathrm{JAC3}]_e \tag{6.36}$$

where $\otimes$ denotes the matrix tensor product, a matrix multiplication between a rank α and rank β matrix producing a rank $(\alpha+\beta)$ matrix. The TP matrix factorization assumption is that each factor in (6.36) is of the form

$$[\mathrm{JAC}I]_e = [\mathrm{M}I]_e + \theta\Delta t\big([\mathrm{V}I]_e + [\mathrm{D}I]_e + [\mathrm{DB}I]_e + \cdots\big) \tag{6.37}$$

for I an integer $(1, \ldots, d)$. Each of the matrices defined in (6.37) is assumed formed using d=1 FE bases, on one-dimensional domains, hence is of rank k+1. Referring to Table 5.2, the lead matrix in (6.36) in direction 1 is

$$[\mathrm{M1}]_e = det_e[\mathrm{A200}] = \frac{\ell_1}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{6.38}$$

where ℓ_1 is the element length parallel to η_1. The corresponding term parallel to η_2 is then

$$[\mathrm{M2}]_e = det_e[\mathrm{A200}] = \frac{\ell_2}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{6.39}$$

and the η_3 term form is obvious.

The matrix tensor product of (6.37)-(6.38) is

$$[\mathrm{M1}]_e \otimes [\mathrm{M2}]_e = \frac{l_1}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \otimes \frac{l_2}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} = \frac{\ell_1\ell_2}{36}\begin{bmatrix} 4 & 2 & 1 & 2 \\ 2 & 4 & 2 & 1 \\ 1 & 2 & 4 & 2 \\ 2 & 1 & 2 & 4 \end{bmatrix} = det_e[\mathrm{B200}] \tag{6.40}$$

i.e., it is identical to the d=2 mass matrix given in Table 5.2. By direct extension, then

$$[\mathrm{M1}]_e \otimes [\mathrm{M2}]_e \otimes [\mathrm{M3}]_e = det_e[\mathrm{C200}] \tag{6.41}$$

hence the matrix tensor product operation can generate multi-dimensional matrices from d=1 forms. In fact, it is easy to verify in η_j space that

$$[\mathrm{M1}]_e \otimes [\mathrm{V2}]_e = \mathrm{ET22}_e[\mathrm{B202}] \tag{6.42}$$

since $\mathrm{ET21}_e$=0 for x_i and η_j parallel. Thereby also, for example

$$[\mathrm{M1}]_e \otimes [\mathrm{M2}]_e \otimes [\mathrm{V3}]_e = \mathrm{ET33}_e[\mathrm{C203}] \tag{6.43}$$

and the remaining tensor products indeed produce multi-dimensional element matrices from d=1 constructions in η_j space.

Substituting (6.37) into (6.36), restricting to d=2, deleting $[\mathrm{DB}I]_e$ for clarity and using (6.38)-(6.42) as appropriate, yields

$$\begin{aligned}
[\mathrm{JAC}]_e &\cong [\mathrm{JAC1}]_e \otimes [\mathrm{JAC2}]_e \\
&= [\mathrm{M1}]_e \otimes [\mathrm{M2}]_e + [\mathrm{M1}]_e \otimes \theta\Delta t\left([\mathrm{V2}]_e + [\mathrm{D2}]_e\right) \\
&\qquad + [\mathrm{M2}]_e \otimes \theta\Delta t\left([\mathrm{V1}]_e + [\mathrm{D1}]_e\right) \\
&\qquad + (\theta\Delta t)^2\left([\mathrm{V1}]_e \otimes [\mathrm{V2}]_e + [\mathrm{D1}]_e \otimes [\mathrm{D2}]_e + \cdots\right) \\
&= [\mathrm{M}]_e + \theta\Delta t\left([\mathrm{V}]_e + [\mathrm{D}]_e\right) + (\theta\Delta t)^2(\mathrm{error}))
\end{aligned} \tag{6.44}$$

Therefore, the matrix tensor product of a sum of element matrices reconstructs the matrices in the parent expression with the addition of a error term of order $(\theta\Delta t)^2$. This error is the same order as the developed ODE methods, and is not of consequence if the quasi-Newton algorithm is iterated to convergence. However, if the mesh is poor enough, then this error term can adversely affect convergence (rate), hence also iterative stability.

This discussion introduces the mechanics of the TP operation. The purpose is not multi-dimensional element matrix construction, however, but rather to establish an efficient quasi-Newton jacobian replacement for (3.17a), the assembly of (6.34). In global terms then, (3.17a) is approximated as

$$[\mathrm{JAC}]\{\delta Q\} \cong [\mathrm{JAC1}] \otimes [\mathrm{JAC2}] \otimes [\mathrm{JAC3}]\{\delta Q\}^{p+1} = -\{FQ\}^p \tag{6.45}$$

For the definitions [JAC3] $\{\delta Q\}^{p+1} \equiv \{S\}$ and [JAC2] $\otimes$ [JAC3] $\{\delta Q\}^{p+1} \equiv \{R\}$, *then the TP* factorization quasi-Newton procedure is the *d*-step sequence

$$
\begin{aligned}
[\mathrm{JAC1}]\{R\} &= -\{FQ\}^{p} \\
[\mathrm{JAC2}]\{S\} &= -\left\{R^{T}\right\} \\
[\mathrm{JAC3}]\left\{\delta\hat{Q}\right\}^{p+1} &= \left\{S^{T}\right\}
\end{aligned}
\tag{6.46}
$$

The superscript "hat" emphasizes that the solution to (6.46) is only an approximation to $\{\delta Q\}^{p+1}$, and the intermediate data arrays $\{R\}$ and $\{S\}$ must be row-column exchanged between generation and use, hence the (internal) superscript "*T*."

The linear algebra procedure (6.46) thus operates as a nodal string sweeping procedure, using block-($2k$+1) diagonal matrices to replace the large sparse Newton jacobian (3.17a). Thus, an underlying assumption is existence of a lexicographic ordering, such that data strings amenable to efficient handling exist. There is no statement in the derivation that restricts this ordering to uniform, i.e., the domain Ω transforms to the "unit box." However, in this project as well as in efforts following, we have not attempted to seriously explore other than simply ordered lexigographics. The real attraction of (6.45), as a linear algebra procedure, is that well established avenues to massive parallelism exist, which is the subject of ongoing research, c.f., Manning, *et al* (1993).

To the specific issue of this project, the goal is to enable use of "*arbitrary meshing*" which is interpreted as one exhibiting little (no) restriction on distortion in the field. Specifically, a regular (rectangular cartesian) meshing is not admissible, hence the derived TP algorithm (6.45) must be adapted to handle arbitrarily non-regular meshings. Following considerable study (Iannelli, 1991), the required TP generalization reduced to accurate metric data handling and elimination of the factor det_e on the [A200] mass matrix. For the latter, the Newton algorithm is recast to include a multi-dimensional nodal *det*={DET} into the iteration variable, such that (3.16) becomes

$$[JAC]\{\mathrm{DET}\delta Q\}^{p+1} = -\{FQ\}^{p} \tag{6.47}$$

$$[JAC]_e \equiv \frac{\partial\{FQ\}_e}{\partial\{\mathrm{DETQ}\}_e} = \frac{\partial}{\partial\{\mathrm{DETQ}\}_e}\left([M]\left\{Q^{p}_{n+1} - Q_n\right\} + \Delta t\{RQ\}_{n+\theta}\right)_e \tag{6.48}$$

which would involve numerous modifications to algorithm jacobian derivations in Section 5.4. In practice, the formulation change is minimal via interior multiplication in (6.47) by $[I]=\lceil DET \rfloor^{-1} \lceil DET \rfloor$ yielding

$$[JAC]\lceil DET \rfloor^{-1} \lceil DET \rfloor \{\delta Q\} = [JAC] \lceil DET \rfloor^{-1} \{DET\delta Q\} \tag{6.49}$$

where $\lceil DET \rfloor$ is diagonal matrix form of $\{DET\}$. Hence, each jacobian contribution in Section 5.4 receives only a post multiplier $\lceil DET \rfloor_e^{-1}$, and upon solution of (6.47)

$$\{\delta Q\}^{p+1} = [DET]^{-1}\{DET\delta Q\} \tag{6.50}$$

The second key ingredient for TP factorization to function accurately on arbitrarily nonregular meshings is to replace elementary length scales with the full multi-dimensional metric data distributions, recall (5.21). Iannelli(1991, pg. 207-280) exhaustively examines this issue, hence derives the functional forms for the TP jacobians $[JACI]_e$. The implementation employs the developed metric tensor data base TIJ_e in concert with [A...] element matrices and the array of inverse nodal determinant as the diagonal matrix post-multiplication given in (6.49) at the element level. For example, (5.59) defines the self-coupled Newton jacobian total energy E contribution from the residual in compact form as

$$\begin{aligned}[RE,E]_e = &-\{UJ\}_e^T[M30J0]_e + (\gamma-1)\{UJ\}_e^T[M30J0]_e \\ &+ \frac{(\gamma-1)Ec}{PeEu}[M2KK]_e\lceil OSRH \rfloor_e + \beta_E \hat{U}K_e\{UJ\}_e^T[M30KJ]_e\end{aligned} \tag{6.51}$$

The expanded form for this jacobian, including the mass matrix term, boundary conditions, and all metric data detail is presented in Fig. 6.1, as excerpted from Appendix A, the d=2 *AKCESS*.AERO template equivalent of (6.51) with $[M]_e$ from (5.12). The entries are grouped by contributions as: (1 1) is the time term $[M]_e$=DET$_e$[B200]; (2 1) is (6.51) with GM1=(γ-1) and ZPEC and HBM are scalar switches; (3 1) is the inflow BC set with switch PHBI; (4 1) is the companion outflow BC set with switch PHBO. The bracket distribution of data is defined in (5.31), and recall the group preceding a master matrix (B...) contains the metric data distribution and det_e exponent. Empty brackets remain interpreted as unity, except in the final location which is the identity diagonal matrix.

```
JACOBIANS

  ETOT ETOT   1 1   T
 ()()()(;1)(B200)()
  ETOT ETOT   2 1   T
+(-)()(U1+U2)(102;0)(B3010)()+(-)()(U1+U2)(304;0)(B3020)()
+(-GM1)()(U1+U2)(102;0)(B3010)()
+(-GM1)()(U1+U2)(304;0)(B3020)()
+(ZPEC,PEI)()()(1122;-1)(B211)()+(ZPEC,PEI)()()(3344;-1)(B222)()
+(ZPEC,PEI)()()(1324;-1)(B221)()+(ZPEC,PEI)()()(1324;-1)(B212)()
+(HBM)()(UMAG,HB1)(1122;-1)(B3011)()+(HBM)()(UMAG,HB1)(3344;-1)(B3022)()
+(HBM)()(UMAG,HB1)(1324;-1)(B3012)()+(HBM)()(UMAG,HB1)(1324;-1)(B3021)()
  ETOT ETOT   3   1   T
 ()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+(GM1)()(U1+U2)(102;0)(B3010)()+(GM1)()(U1+U2)(102;0)(B31P1)()
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(GM1)()(U1+U2)(304;0)(B3020)()+(GM1)()(U1+U2)(304;0)(B32P2)()
+(PHBI)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHBI)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHBI)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHBI)()(UMAG,HB1)(1324;-1)(B3021)()
  ETOT ETOT   4   1   T
 ()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+(GM1)()(U1+U2)(102;0)(B3010)()+(GM1)()(U1+U2)(102;0)(B31P1)()
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(GM1)()(U1+U2)(304;0)(B3020)()+(GM1)()(U1+U2)(304;0)(B32P2)()
+(PHBO)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHBO)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHBO)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHBO)()(UMAG,HB1)(1324;-1)(B3021)()
```

Figure 6.1 *AKCESS*.AERO *REMI* template jacobian $[RE,E]_e$, d=2.

Figure 6.2 contains the comparison *AKCESS*.AERO template instruction set for the TP factorized quasi-Newton jacobian for d=2 problems. Hence, there are two jacobians defined, and each contain the four cited groupings. Comparing Fig. 6.2 to Fig. 6.1, in the first group (1 1) note that (A200) is post multiplied by DETJ, the diagonal matrix of $(1/DET)_{node}$. In the preceding bracket, the "O" following the semi-colon is the exponent on DET_e, replacing the (; 1) in Fig. 6.1. In Fig. 6.2, the first line in group (2 1) is the convection term, and therein (102;0) preceding (A3010) denotes a d-dimensional $(ET11+ET21)_e$ multiplication to establish the *contravariant* velocity resolution parallel to the η_1 direction. Skipping down to the second factored jacobian, these metric data become (304;0) for the η_2-direction contravariant component.

The other data in Fig. 6.2 or Fig. 6.1 are clearly "readable," once the syntax and notational structure are understood. Hence, the *template* construct for TWS^h FE algorithms via *AKCESS*.* provides a precise definition for *every nuance* of an algorithm. Appendix C contains the *AKCESS*.AERO *template* for the complete d=2 TP quasi-Newton jacobian construction.

```
JACOBIANS
  #
  #  FACTORED JACOBIAN FOR DIRECTION 1  #
  #
  ETOT ETOT  1   1  #
 ()()()(;0)(A200)(DETJ)
  ETOT ETOT  2   1  #
 (-)()(U1+U2)(102;0)(A3010)()
+(-GM1)()(U1+U2)(102;0)(A3010)()
+(PDUM2,PEI)()()(1122;-1)(A211)()
+(HBM)(U1,UMGB)(U1)(11;-0.5)(A3011)()
+(HBM)(U2,UMGB)(U2)(22;-0.5)(A3011)()
  ETOT ETOT  3  1  #
 ()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(GM1)()(U1+U2)(102;0)(A3010)()+(GM1)()(U1+U2)(102;0)(A31P1)()
+(PHBI)(U1,UMGB)(U1)(11;-0.5)(A3011)()
+(PHBI)(U2,UMGB)(U2)(22;-0.5)(A3011)()
  ETOT ETOT  4  1  #
 ()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(GM1)()(U1+U2)(102;0)(A3010)()+(GM1)()(U1+U2)(102;0)(A31P1)()
+(PHBO)(U1,UMGB)(U1)(11;-0.5)(A3011)()
+(PHBO)(U2,UMGB)(U2)(22;-0.5)(A3011)()
#
#  FACTORED JACOBIAN FOR DIRECTION 2  #
#
  ETOT ETOT  1   2  #
 ()()()(;0)(A200)(DETJ)
  ETOT ETOT  2   2  #
 (-)()(U1+U2)(304;0)(A3010)()
+(-GM1)()(U1+U2)(304;0)(A3010)()
+(PDUM2,PEI)()()(3344;-1)(A211)()
+(HBM)(U1,UMGB)(U1)(33;-0.5)(A3011)()
+(HBM)(U2,UMGB)(U2)(44;-0.5)(A3011)()
  ETOT ETOT  3  2  #
 ()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(GM1)()(U1+U2)(304;0)(A3010)()+(GM1)()(U1+U2)(304;0)(A31P1)()
+(PHBI)(U1,UMGB)(U1)(33;-0.5)(A3011)()
+(PHBI)(U2,UMGB)(U2)(44;-0.5)(A3011)()
  ETOT ETOT  4  2  #
 ()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(GM1)()(U1+U2)(304;0)(A3010)()+(GM1)()(U1+U2)(304;0)(A31P1)()
+(PHBO)(U1,UMGB)(U1)(33;-0.5)(A3011)()
+(PHBO)(U2,UMGB)(U2)(44;-0.5)(A3011)()
```

Figure 6.2 *AKCESS*.AERO *RE*MI template, TP jacobian [RE,ETP], d=2

6.6 Summary

This section has established several auxiliary procedures applicable to the TWS^h *REMI* CFD algorithm. The matrix tensor product quasi-Newton approximation for linear algebra efficiency is derived and clearly established as an ***AKCESS.****** template. This completes derivation and definition of the TWS^h aerodynamics CFD algorithm.

7. DISCUSSION AND RESULTS

7.1 Synopsis

During the specified performance period of this contract, only rudimentary verification-type computational results were accomplished, due mainly to a primitive code base. During the subsequent two years of formal extension, *AKCESS.** began to function in parallel with dissertation project development of the FEMNAS code (Iannelli, 1991). The intervening period has seen maturation of *AKCESS.**, which has finally provided the software platform to critically assess TWS^h algorithm issues that were "hard-wired" one way in the FEMNAS code. This section discusses verification, benchmark and validation computational experiment results for the developed *REMI* FE CFD theory for aerodynamics applications.

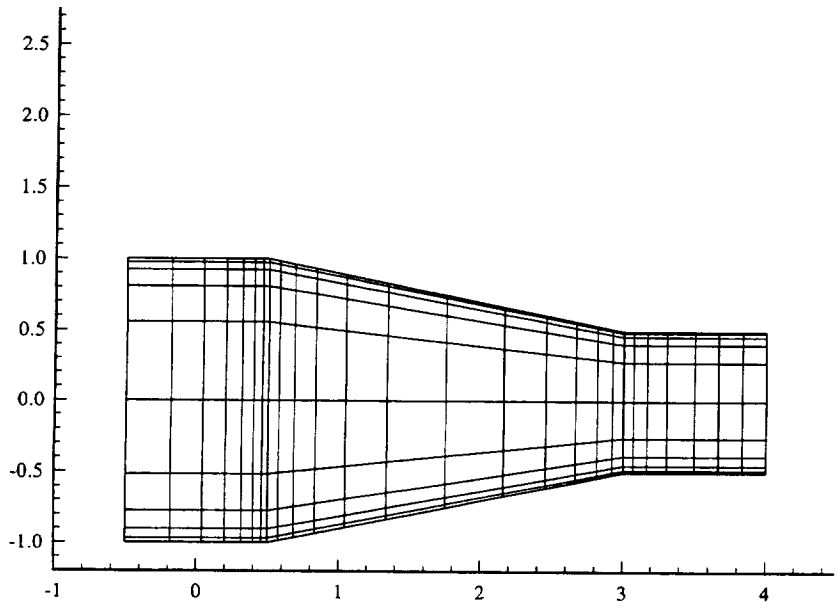
7.2 Subsonic inviscid verifications, d=2,3

The emergence of *AKCESS*.AERO has finally provided the platform to critically assess TWS^h algorithm issues of stability, mesh non-uniformity and Newton versus tensor product (TP) quasi-Newton matrix solution methods. The verification problem is shock-free subsonic internal flow in a duct of varying cross-section. Figure 7.1 graphs a 2:1 area ratio duct for d=2, and the companion 4:1 area ratio d=3 duct. For d=2, a range of highly non-uniform meshings are employed to assess "mesh arbitrariness," while for d=3 the transformation of hexahedra to a circular cross-section introduces large mesh skewing at the 45° rays as illustrated.

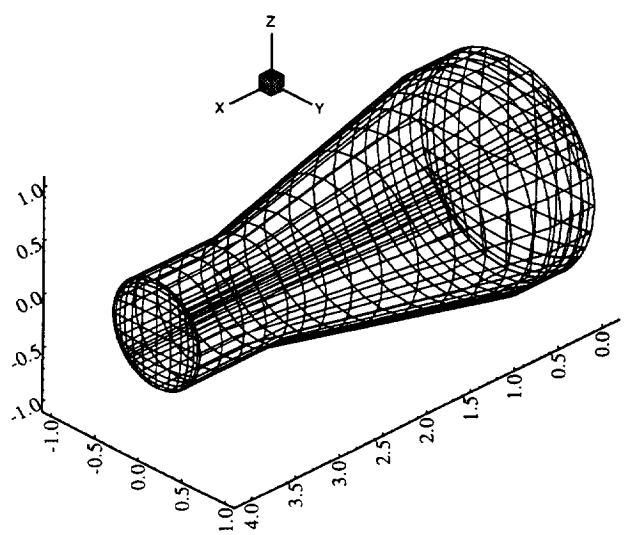
Prior to the *REMI* state variable IC process, verification problems required about a hundred iterations to self-generate a tangency-adequate solution field. With the IC algorithm, Section 6.2, the *REMI* solution process proceeds immediately. Three distinct *AKCESS.** templates for the d=2 IC algorithm are developed, differing by distributed nodal $[RHO]_e$ vs. element-average RHO_e, template-discernible by the bracket where ρ_e appears, and the use of [B30*JK*] hyper matrices in the former, Fig. 7.2a). This template also uses element-average metric data, since no suffix "E" appears in the FE matrix name. Conversely, Fig. 7.2b) is the d=2 Gauss-quadrature template, distinguished by the element matrices [B2*JK*E], with the last integer in the metric data string ("003" in this case) the symmetric order (P=Q) of the quadrature rule, recall (5.17). Finally, the d=3 Gauss quadrature template is listed in Fig. 7.3, distinguished by the element matrix prefix "C" and P=3=Q.

A mesh distortion sensitivity study is performed for the d=2 case via gross displacement of select nodes of the *macro mesh*, i.e., the coarsest mesh that defines the

PRECEDING PAGE BLANK NOT FILMED



a)



b)

Figure 7.1 Converging subsonic duct verification, a) d=2, 2:1 area ratio, b) d=3, 4:1 area ratio.

```
TITLE                    **TEMPLATE FILE TEMP.B30KJ****                    a)
                    REMI.IC 2D, NODAL RHO, DELSQ PHI (1/04/94)
RESIDUALS
 PHI 2       T    VARBL,      SET NO., --- SPATIAL          SET (PHI)
 ()()(RHO)(1122;-1)(B3011)(PHI)+()()(RHO)(3344;-1)(B3022)(PHI)
+()()(RHO)(2413;-1)(B3021)(PHI)+()()(RHO)(1324;-1)(B3012)(PHI)
 PHI 4       T    VARBL,      SET NO., --- BOUNDARY        SET (PHI)
 ()()()(;1)(A200)(UJNJ)

JACOBIANS
 PHI  PHI  2      1  T      VARBL, VARDIF,     SET,          NEWTON
 ()()(RHO)(1122;-1)(B3011)()+()()(RHO)(3344;-1)(B3022)()
+()()(RHO)(2413;-1)(B3021)()+()()(RHO)(1324;-1)(B3012)()

TITLE                    *** TEMPLATE TEMP.B30KJ ***
                    2D WS VELOCITY CALCULATION      (1/04/94)
RESIDUALS
 U1       2   T    VARBL,      SET NO., --- DATA  SET (U1)
 ()()()(1;0)(B201)(PHI)
+()()()(3;0)(B202)(PHI)
 U2       2    T   VARBL,      SET NO., --- DATA  SET (U2)
 ()()()(2;0)(B201)(PHI)
+()()()(4;0)(B202)(PHI)

JACOBIANS
 U1  U1      2  1      T VARBL,    VARDIF,     SET NO.     JACOBIAN (U1)
()()()(0;1)(B200)()
 U2  U2      2  1      T VARBL,    VARDIF,     SET NO.     JACOBIAN (U2)
()()()(0;1)(B200)()

TITLE                    *** TEMPLATE FILE TEMP.B2KJE ***                  b)
                    REMI.IC 2D, AVE. RHO,  DELSQ PHI SOLVE  (1/04/94)
RESIDUALS
 PHI 2       T                        VARBL,      SET NO,     SPATIAL SET (PHI)
 ()(RHO)()(1122003;-1)(B211E)(PHI)+()(RHO)()(3344003;-1)(B222E)(PHI)
+()(RHO)()(2413003;-1)(B221E)(PHI)+()(RHO)()(1324003;-1)(B212E)(PHI)
 PHI 4   T  VARBL, SET NO., --- BOUNDARY SET (PHI)
 ()()()(;1)(A200)(UJNJ)

JACOBIANS
 PHI     PHI      2   1    T        VARBL, VARDIF,      SET,      NEWTON
 ()(RHO)()(1122003;-1)(B211E)()+()(RHO)()(3344003;-1)(B222E)()
+()(RHO)()(2413003;-1)(B221E)()+()(RHO)()(1324003;-1)(B212E)()

TITLE                    *** TEMPLATE TEMP.B2KJE ***
                    2D WS VELOCITY CALCULATION      (1/04/94)
RESIDUALS
 U1          2          T    VARBL,      SET NO., --- DATA SET (U1)
 ()()()(1000003;0)(B201E)(PHI)
+()()()(3000003;0)(B202E)(PHI)
 U2  2                  T  VARBL,       SET NO., --- DATA SET (U2)
 ()()()(2000003;0)(B201E)(PHI)
+()()()(4000003;0)(B202E)(PHI)

JACOBIANS
 U1  U1      2    1    T    VARBL, VARDIF, SET NO., JACOBIAN (U1)
()()()(0000003;1)(B200E)()
U2  U2      2    1    T    VARBL, VARDIF, SET NO., JACOBIAN (U2)
()()()(0000003;1)(B200E)()
```

Figure 7.2 *AKCESS.** template for *REMI d*=2 IC generation, a) nodal density, element-averaged metric data b) average density, Gauss quadrature element matrices.

```
TITLE                    **** TEMPLATE FILE TEMP.C2KJE ****
                   REMI.IC 3D, AVG. RHO,  DELSQ PHI (1/06/94)

RESIDUALS
 PHI 2  T  VARBL, SET NO., --- SPATIAL SET (PHI)
 ()()(RHO)(1122333;-1)(C211E)(PHI)+()()(RHO)(4455663;-1)(C222E)(PHI)
+()()(RHO)(7788993;-1)(C233E)(PHI)+()()(RHO)(1425363;-1)(C212E)(PHI)
+()()(RHO)(1425363;-1)(C221E)(PHI)+()()(RHO)(1728393;-1)(C213E)(PHI)
+()()(RHO)(1728393;-1)(C231E)(PHI)+()()(RHO)(4758693;-1)(C223E)(PHI)
+()()(RHO)(4758693;-1)(C232E)(PHI)

 PHI 4   T  VARBL, SET NO., --- BOUNDARY SET (PHI)
 ()()()(;1)(B200)(UJNJ)

JACOBIANS
 PHI PHI 2 1 T VARBL, VARDIF, SET NO., JACOBIAN (PHI)
 ()()(RHO)(1122333;-1)(C211E)()+()()(RHO)(4455663;-1)(C222E)()
+()()(RHO)(7788993;-1)(C233E)()+()()(RHO)(1425363;-1)(C212E)()
+()()(RHO)(1425363;-1)(C221E)()+()()(RHO)(1728393;-1)(C213E)()
+()()(RHO)(1728393;-1)(C231E)()+()()(RHO)(4758693;-1)(C223E)()
+()()(RHO)(4758693;-1)(C232E)()

TITLE                    **** TEMPLATE TEMP.C2KJE ****
                   3DWS VELOCITY CALCULATION (1/06/94)

RESIDUALS
 U1   2   #  VARBL, SET NO., --- DATA SET (U1)
 ()()()(1000003;0)(C201E)(PHI)
+()()()(4000003;0)(C202E)(PHI)
+()()()(7000003;0)(C203E)(PHI)

 U2   2  T  VARBL, SET NO., --- DATA SET (U2)
 ()()()(2000003;0)(C201E)(PHI)
+()()()(5000003;0)(C202E)(PHI)
+()()()(8000003;0)(C203E)(PHI)

 U3   2  T  VARBL, SET NO., --- DATA SET (U3)
 ()()()(3000003;0)(C201E)(PHI)
+()()()(6000003;0)(C202E)(PHI)
+()()()(9000003;0)(C203E)(PHI)

JACOBIANS
 U1  U1  2  1 T VARBL, VARDIF, SET NO., JACOBIAN (U1)
 ()()()(0;1)(C200)()

 U2  U2  2  1 T VARBL, VARDIF, SET NO., JACOBIAN (U2)
 ()()()(0;1)(C200)()

 U3  U3  2  1 T VARBL, VARDIF, SET NO., JACOBIAN (U3)
 ()()()(0;1)(C200)()
```

Figure 7.3 *AKCESS.** template for *REMI* d=3 IC generation, averaged density, Gauss quadrature element matrices.

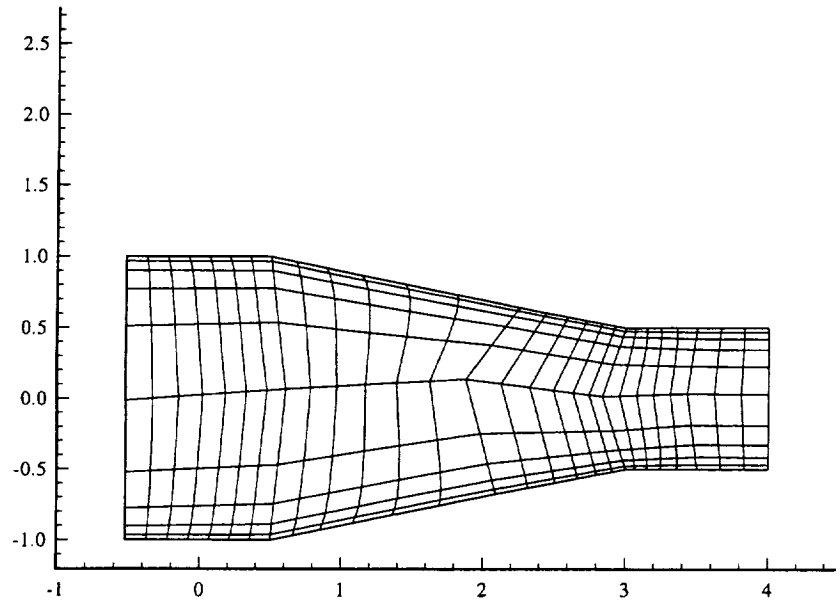
geometric essence and admits local mesh adaptation/refinement as needed. Figure 7.4 shows two levels of mesh distortion, accomplished by a "click and drag" of two macro vertex nodes, an on-screen, real-time interactive operation. The more distorted mesh possesses nearly singular elements, i.e., ones having an interior angle approximating 180°, yielding a near-triangle, in the interior and near the adverse geometric singularity. To help maintain solution accuracy, the distorted mesh B transverse refinement is double that of mesh A.

Figure 7.5 graphs the IC algorithm converged solutions, for inlet Mach number $Ma \equiv 0.2$, as density distribution with velocity vector field and mesh superimposed using, a)-b) the nodal density, averaged-metric template, and c)-d) the averaged-density, Gauss quadrature template. It is very apparent that averaging metric data on highly distorted meshes is a serious interpolation-error "crime" for subsonic flows (at least). Local departure from density isoclines being nearly vertical lines occurs on both meshings, although velocity vector departures from essential tangency is quite minimal, which verifies robustness for the weak statement algorithm for velocity, recall (6.9). Timing comparisons verify the Gauss quadrature template requires about twice the cpu time to execute, but for this simple case the difference is seconds.

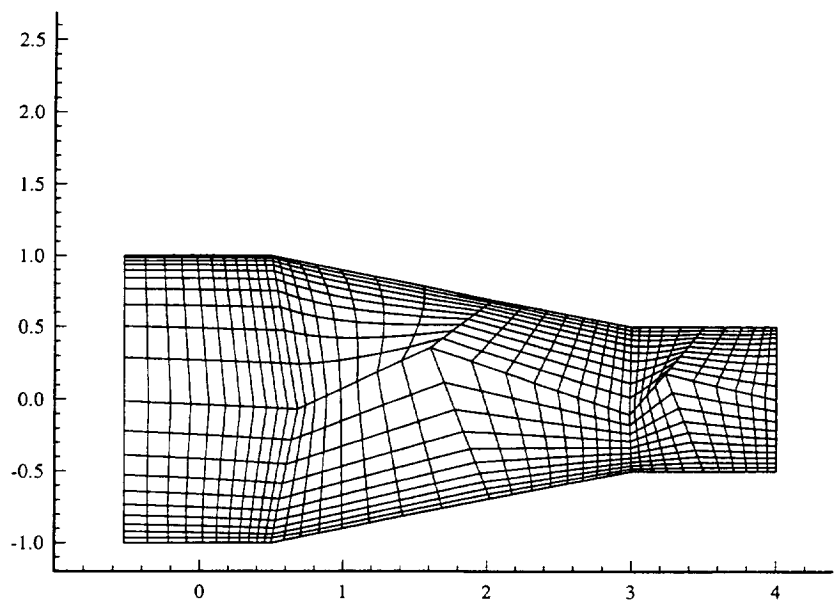
The state variable thus generated serves to initialize a *REMI* Euler equation solution, generated using the scalar β dissipation formulation, c.f. (4.24). Full "upwinding" corresponds to β=1.0, and using β=0.3, the TP *REMI* algorithm (Appendix C) converged to nominal steady-state, from *either* IC data set on the respective meshings, in 100± time steps. Figure 7.6 graphs comparative solutions obtained on mesh B for pressure, with velocity vectors and mesh overlaid, for varying β. Conclusions drawn from these data include:

- averaged-metric data for *REMI* for Euler simulations on quite distorted meshings is admissible
- the TP quasi-Newton factorization is convergent on highly distorted meshes using multi-dimensional metric data
- smooth state variable IC data is not necessary for an Euler solution.
- for the β-range tested, the Euler solution for smooth flows (no shocks) appears relatively unaffected, except near the inflow plane

For the scalar TWS^h dissipation form, decreasing β to 0.10 resulted in solution process divergence on mesh B and marginal instability on the less distorted mesh A. With either smooth or non-smooth IC data, a few dozen TP time steps at β=0.3 readily stablizes an Euler solution, at which point β can be progressively decreased to a minimum level.

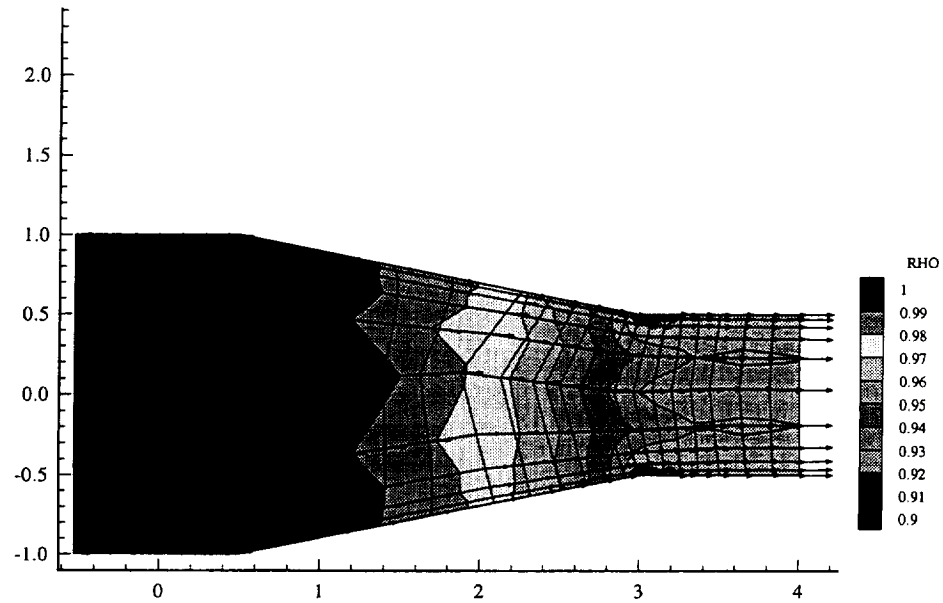


a)

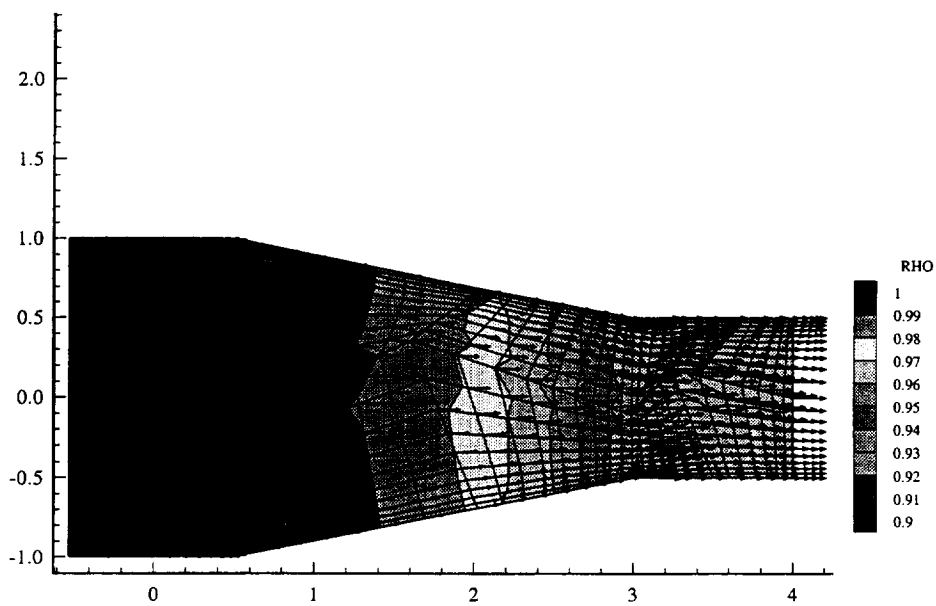


b)

Figure 7.4 Converging duct validation check case, d=2, a) modestly non-cartesian mesh A, b) highly distorted mesh, B.

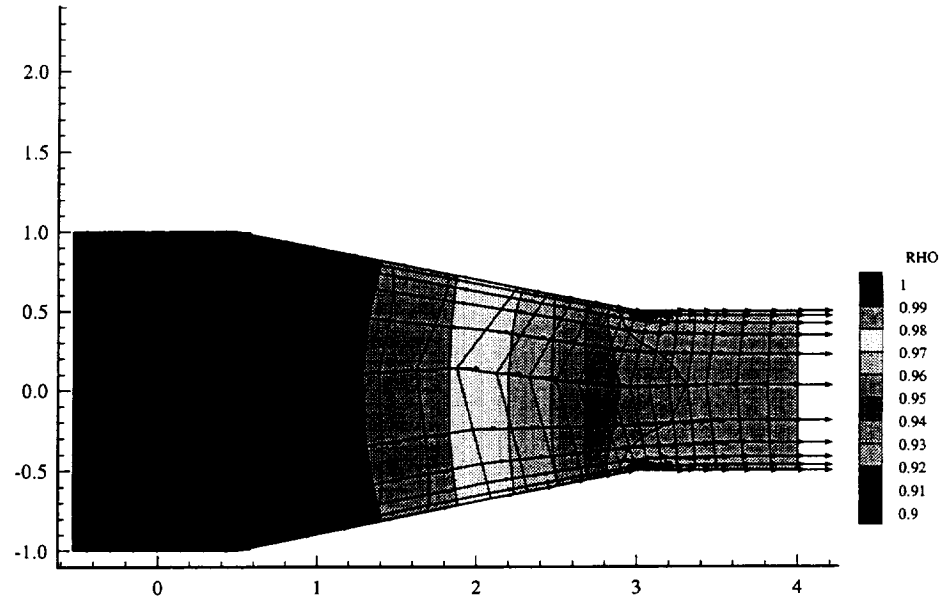


a)

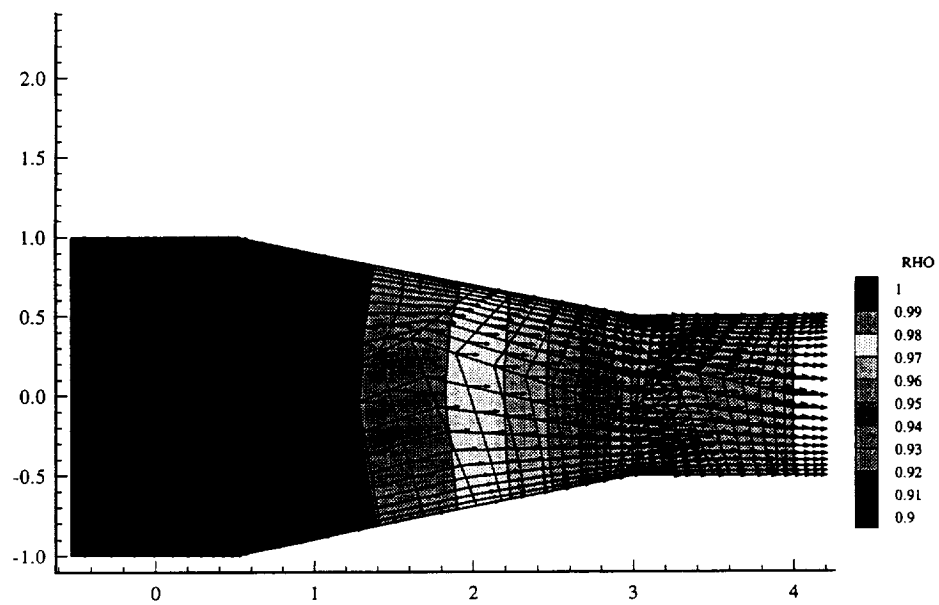


b)

Figure 7.5a IC algorithm TWS^h density solution, d=2 converging duct, $Ma_{in} \cong 0.2$, averaged metric template, a) mesh A, b) mesh B.

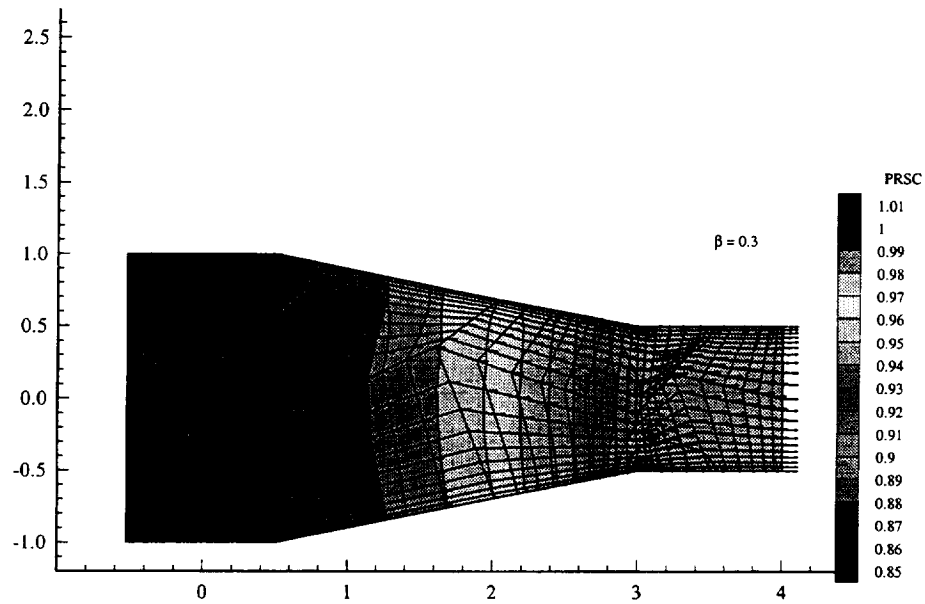


c)

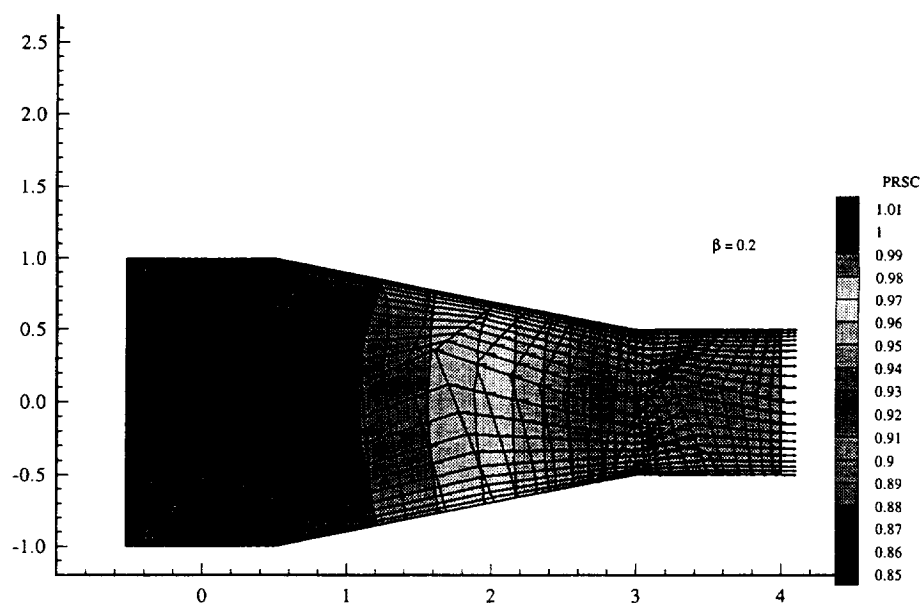


d)

Figure 7.5b IC algorithm TWS^h density solution, d=2 converging duct, $Ma_{in} \cong 0.2$, Gauss quadrature template, c) mesh A, d) mesh B.



a)



b)

Figure 7.6 *REMI* FE TWS^h algorithm Euler steady-state pressure solutions, converging duct, $Ma_{in} \cong 0.2$, scalar β, Mesh B, d=2, a) β=0.3, b) β=0.2

A sensitive measure of error induced by artificial diffusion is loss of total pressure p_T. The non-D definition is

$$p_T = 1 + \gamma \mathrm{Ma}_r^2 \left(m_i m_i / \rho \right) \tag{7.1}$$

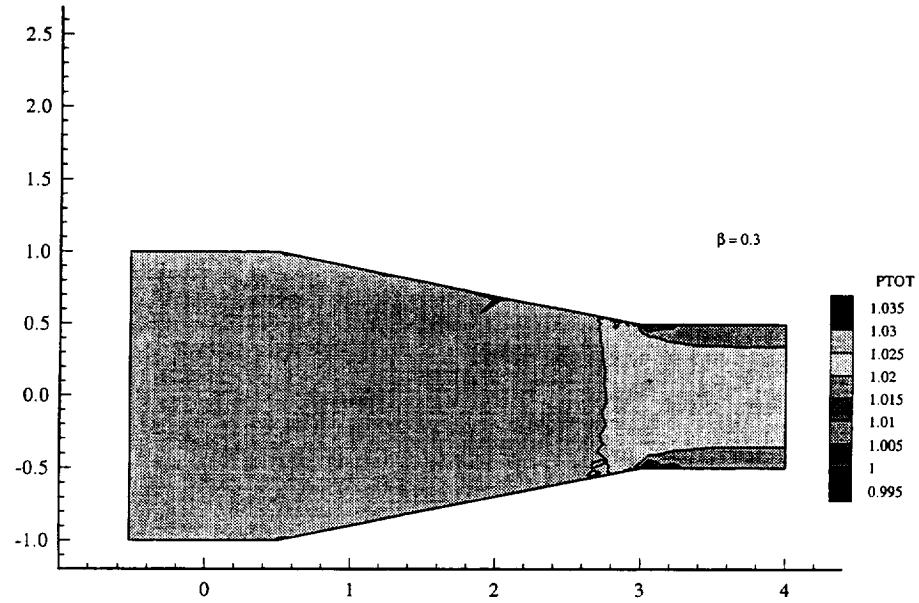
and for inviscid adiabatic duct flow case, p_T should remain nominally equal to the (non-D) stagnation pressure intrinsic to the IC generation process. Figure 7.7 graphs the distribution of p_T for the *REMI* TP Euler solutions on mesh B for scalar β=0.3 and 0.2. The IC generated level is 1.028, which both solutions exhibit everywhere except at locations of mesh coarseness and following the adverse geometric singularity. The lower level of scalar β yields a corresponding smaller p_T loss along outflow walls, as observed by the narrower dark band in the graph.

An Euler solution on distorted mesh B was also executed using the d=2 full Newton jacobian *REMI* template, Appendix A. The allowable Δt for iterative convergence was a factor of about five larger than for the TP algorithm, however the cpu execution time for Newton and TP quasi-Newton was nominally identical. This relates principally to the minimal memory/storage demands for the TP formulation, executed with a Gauss-elimination block-tri-diagonal solver. The full Newton template was solved using the GMRES sparse solver with an incomplete-LU preconditioner, cf., Williams(1993). *AKCESS*.AERO provides these options, with preconditioner variations, as a *template* instruction under "SOLUTION TYPE."

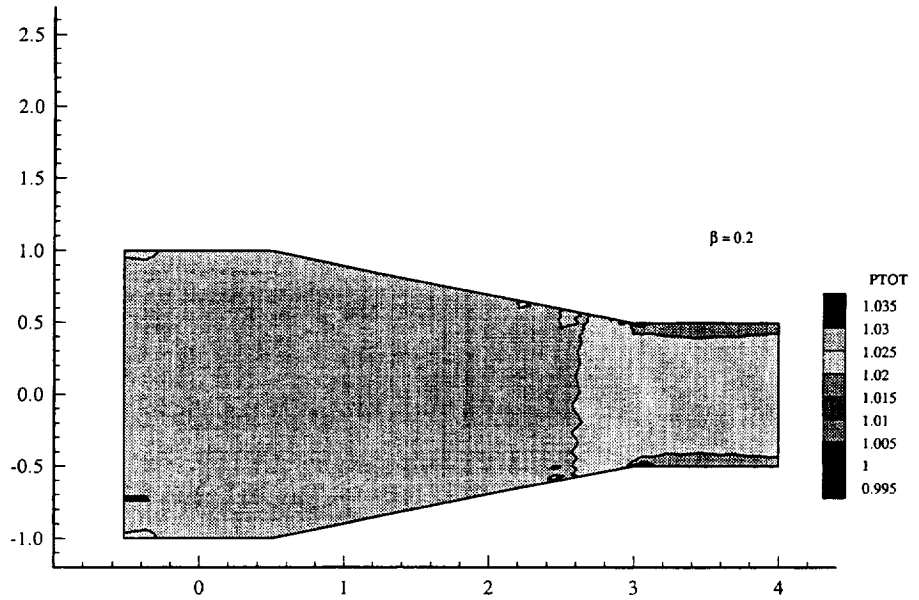
The d=3 *REMI* Newton template is an unattractive option, hence is not constructed. The d=3 TP quasi-Newton jacobian template is given in Appendix D. No artificial internal mesh distortion for IC debug was made, hence sensitivity relates to the relatively poor mesh aspect ratios adjacent to the walls, recall Fig. 7.1. The IC solution generated by the template of Fig. 7.3 is shown in Fig. 7.8a) as density isoclines, which clearly indicates distorted mesh-induced error bands at the four quadrants. However, for scalar β=0.3, the Mach number distributions clearly shows that these IC data errors are annihilated in the Euler solution process, Fig. 7.8b) The exit Mach number distribution does exhibit a modest (2%) variation, however all velocity vectors are clearly normal to the exit plane.

7.3 Transonic inviscid verifications, benchmarks, d=2

The principal purpose of transonic verifications and benchmarks is to assess the shock capturing performance of the FE TWS^h Euler algorithm. A verification problem (with known analytical solution) is flow in a de Laval nozzle, a converging-diverging

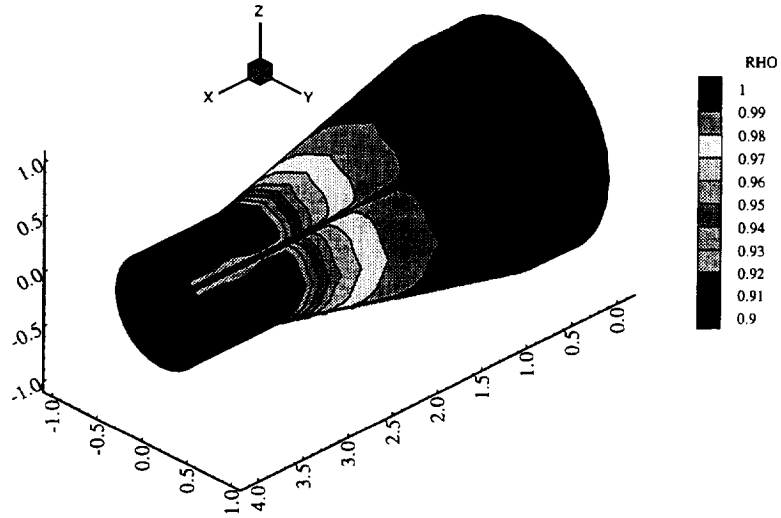


a)

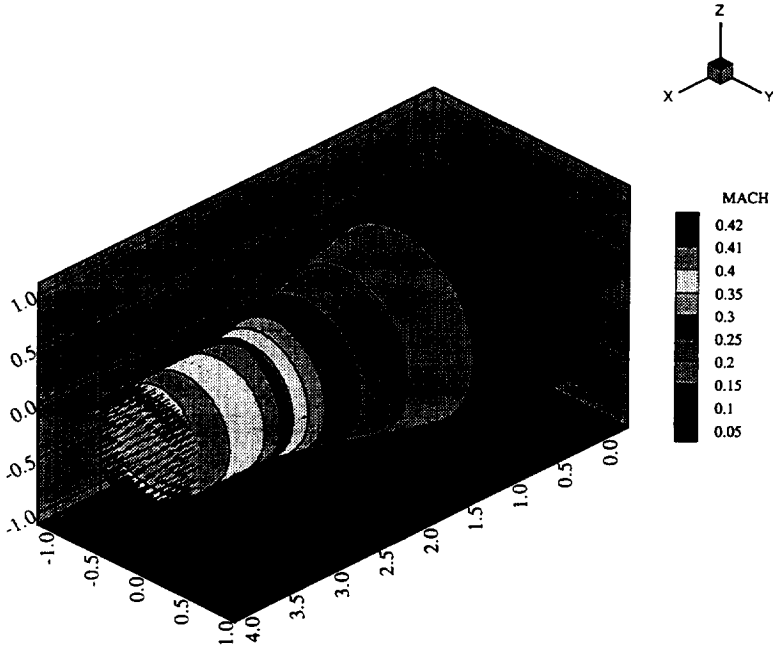


b)

Figure 7.7 Total pressure loss error for *REMI* Euler solutions, converging duct, $Ma_{in} \cong 0.2$, scalar β, d=2, mesh B, d=2, a) β=0.3, b) β=0.2.



a)



b)

Figure 7.8 Converging duct verification, d=3, $Ma_{in} \cong 0.2$, 4:1 area ratio, a) IC density solution, b) *REMI* pressure solution with velocity overlay, scalar β=0.3.

cross-section duct. The flow is axi-symmetric, quasi-one dimensional with governing conservation law form as given in (2.1)-(2. 6), with state variable $q=\{\rho,m,E\}^T$ and with source term array $s=\left\{-m\mathrm{d}(\ln A)/\mathrm{d}x,-\left(m^2/\rho\right)\mathrm{d}(\ln A)/\mathrm{d}x,-(E+p)\mathrm{d}(\ln A)/\mathrm{d}x\right\}^T$, where $A=A(x)$ is the nozzle cross-sectional area distribution. The FEMNAS code generated *REMI* comparative solutions, with θ-Euler and IRK ODE procedures, for the deLaval nozzle geometry of Liou and vanLeer(1988), Fig. 7.9a). The desired inlet Mach number is Ma_{in} =0.24, and p_{out}/p_o = 0.84 yields off-design operation with a normal shock located at x/L=0.65 with shock Mach number Ma_s=1.40. Figure 7.9b) shows the computed steady-state momentum distribution on a uniform 100 element meshing. The TWS^h dissipation parameter set was β_q=0.16 $\{1,1,1\}^T$, i.e., uniform for each state variable member. The shock is clearly located at $x \approx 0.67$ by the discrete approximation to the Rankine-Hugoniot shock relation.

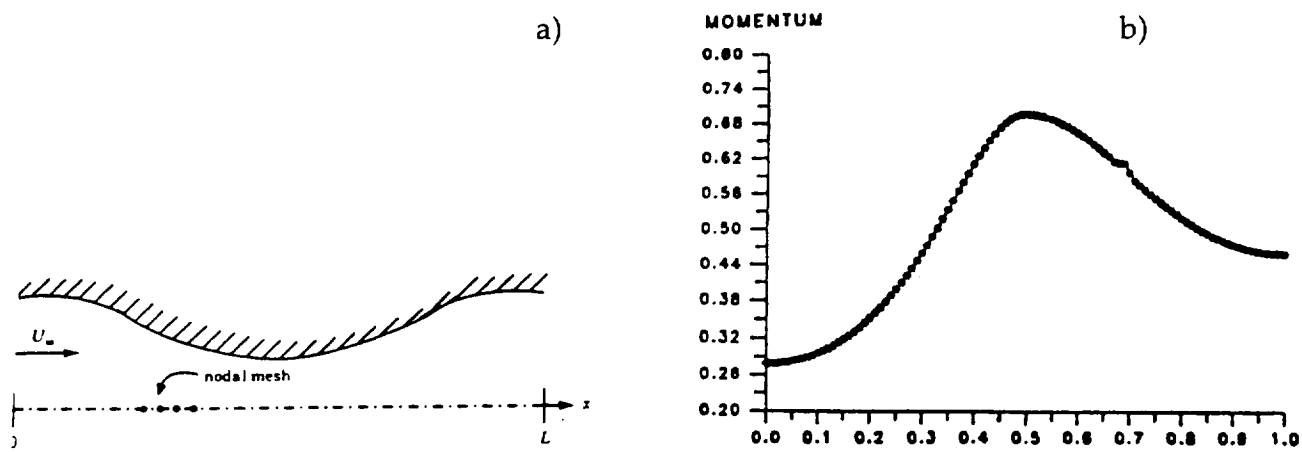


Figure 7.9 deLaval nozzle verification problem, a) cross-section distributions, b) steady *REMI* nodal solution for axial momentum.

Figure 7.10 graphs the unsteady sequence of Mach number distributions computed via *REMI*, using the non-iterative IRK ODE algorithm (6.25)-(6.27) at uniform time step Δt=0.005, which yields $\mathrm{CFL}_{max} \cong 35$, where $\mathrm{CFL}=u\Delta t/\Delta x$. From the initial condition, an expansion wave immediately forms, a), hence moves upstream through the throat, b)-c). Then, the flow downstream of the throat accelerates supersonic, d)-e), to steady-state, f). The resultant shock is absolutely monotone, spread across two FE domains with shock Mach number Ma_s=1.4, in excellent agreement with the analytical solution. Iannelli(1991, Ch. 10.1) details comparative performance for the IRK, the trapezoidal

rule (θ=0.5) and the backwards Euler (θ=1.0) ODE algorithms, for unsteady, time-accurate performance in this geometry. The IRK algorithm exhibited uniformly superior performance for this set of experiments.

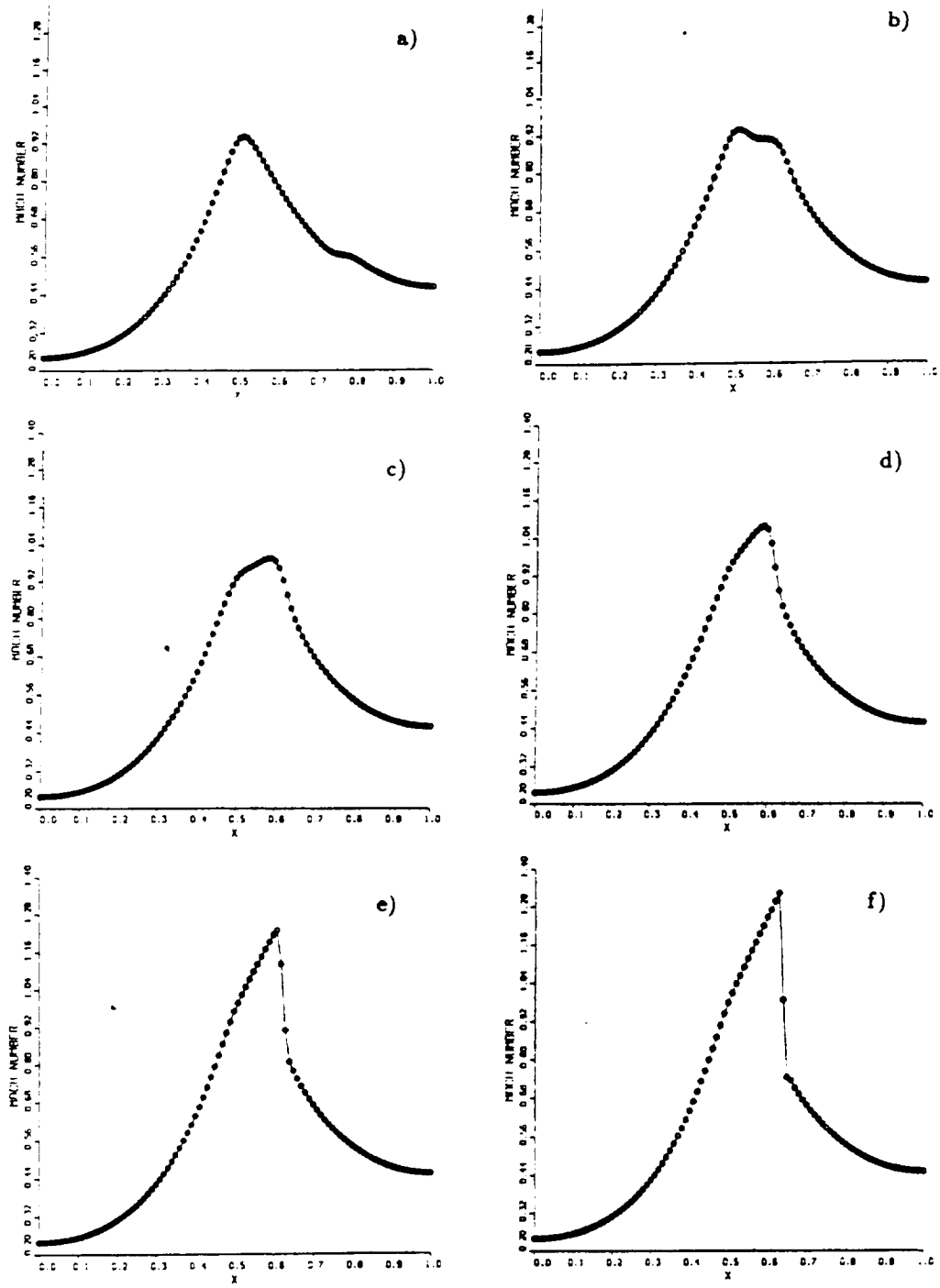


Figure 7.10 deLaval nozzle, unsteady *REMI TWS*^{h} solution for Mach number, IRK ODE algorithm, $\Delta t = 0.005$, a) $t = 0.4$, b) $t = 1.0$, c) $t = 1.2$, d) $d = 1.8$, e) $t = 2.8$.

The d=2 benchmark problem is transonic flow over a 15% thick parabolic arc "airfoil" in a channel. The quasi-one dimensional Euler solution yields a normal shock with Ma_s=1.4, while the d=2 Euler solution predicts a closed supersonic pocket with shock Mach number Ma_s≈1.5, Johnson(1982). Fig. 7.11a) graphs the essential geometry, solution domain, and a nominally uniform 65x35 quad element mesh containing 2376 nodes with modest attractions to the lower wall. The Dirichlet inflow BC are ρ, m_2 and E fixed, the sole fixed BC at outflow is p/p_r =0.68, for which the inlet flow Mach number should approach Ma_{in}=0.68 at steady-state. The top and bottom boundaries are tangent flow with vanishing normal derivatives implied, and the IC was generated via a quasi-one dimensional interpolation procedure.

The steady-state *REMI* FE TWS^h algorithm solution via FEMNAS is summarized in Fig. 7.11-7.12. Figure 7.11b)-d) graphs the steady velocity vector, entropy and Mach number fields; the shock is observed to lie very close to the arc trailing edge, and the Rankine-Hugoniot jump condition is observable in the principal momentum graph, Fig. 7.11e). Fig. 7.12 contains surface perspective and planar isocline presentations for momentum resolution, pressure and Mach number. The shock and Rankine-Hugoniot phenomena are clearly visible, with the shock spread across three elements on the bump surface and Ma_s=1.53, which agrees with Johnson to within 2%. All state variable fields intersect the far-field boundaries in a non-oscillatory manner via the WS^h implied vanishing normal derivative. The impact of relatively coarse mesh at the leading edge is visible as a local oscillation in the transverse momentum graph; otherwise, the *REMI* solution is globally essentially non-oscillatory (ENO).

7.4 Supersonic inviscid verification, d=2

A classic Euler supersonic d=2 verification is flow over a planar wedge with comparison conical analytical solution, cf., Anderson(1982). Figure 7.13a) graphs a uniform 65x35 quad element mesh for wedge angle α=20°. For onset flow at Ma_{in} =3.0, an oblique shock should form at shock angle β=38°, and downstream thereof the flow Mach number is uniform at Ma_{out} =2.0. The entire flowfield is thereby supersonic, hence all state variables are fixed at inflow, and no Dirichlet data is admissible elsewhere. Flow tangency occurs on the stagnation streamline and the wedge surface, and the simulation data should exit smoothly, everywhere downstream of the shock without oscillation. The IC is uniform onset flow interpolated to tangency on the wedge.

This verification test validates the use of FE solution-adaptive remeshing to improve resolution of the oblique shock. Three solution-adapted discretizations were

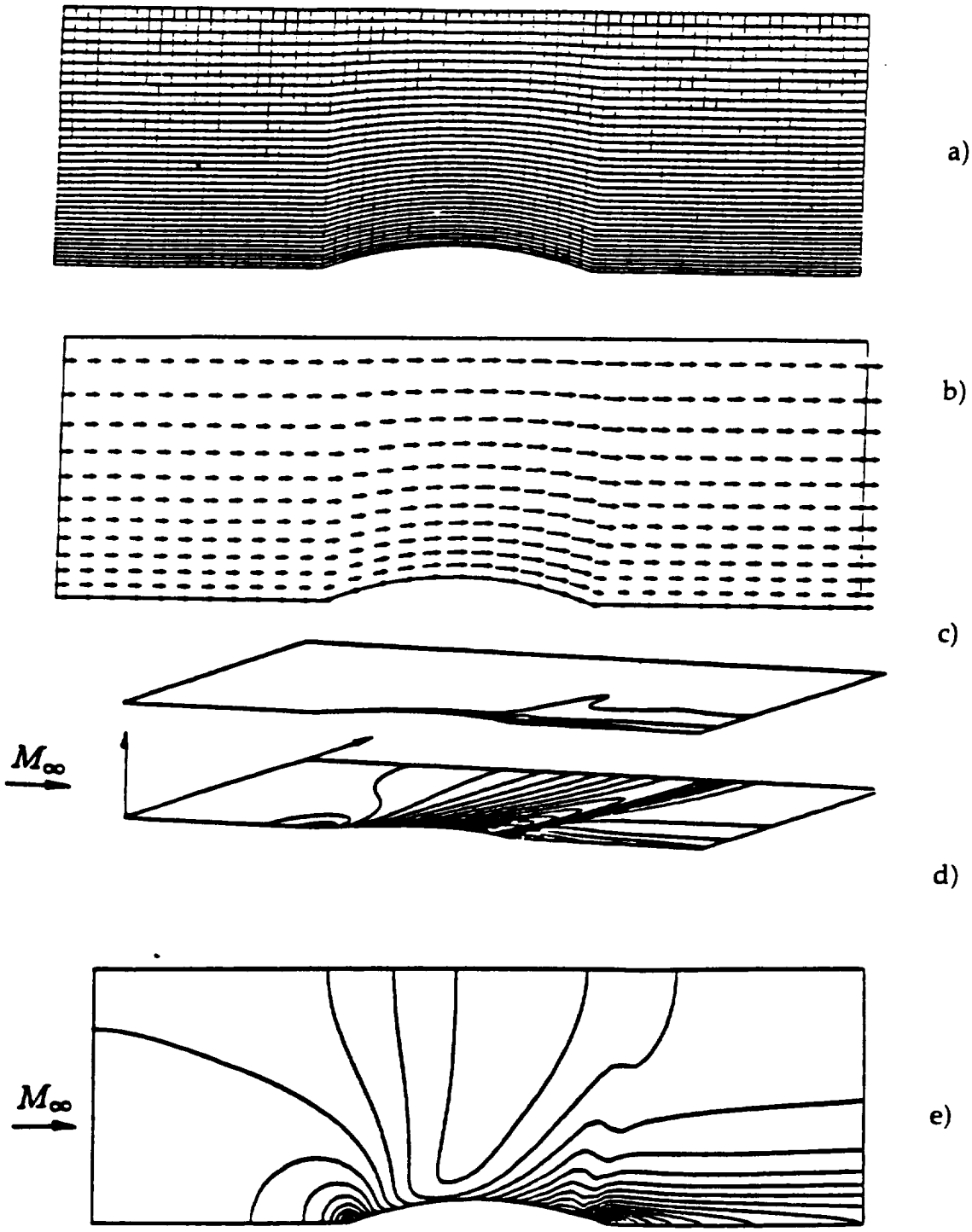
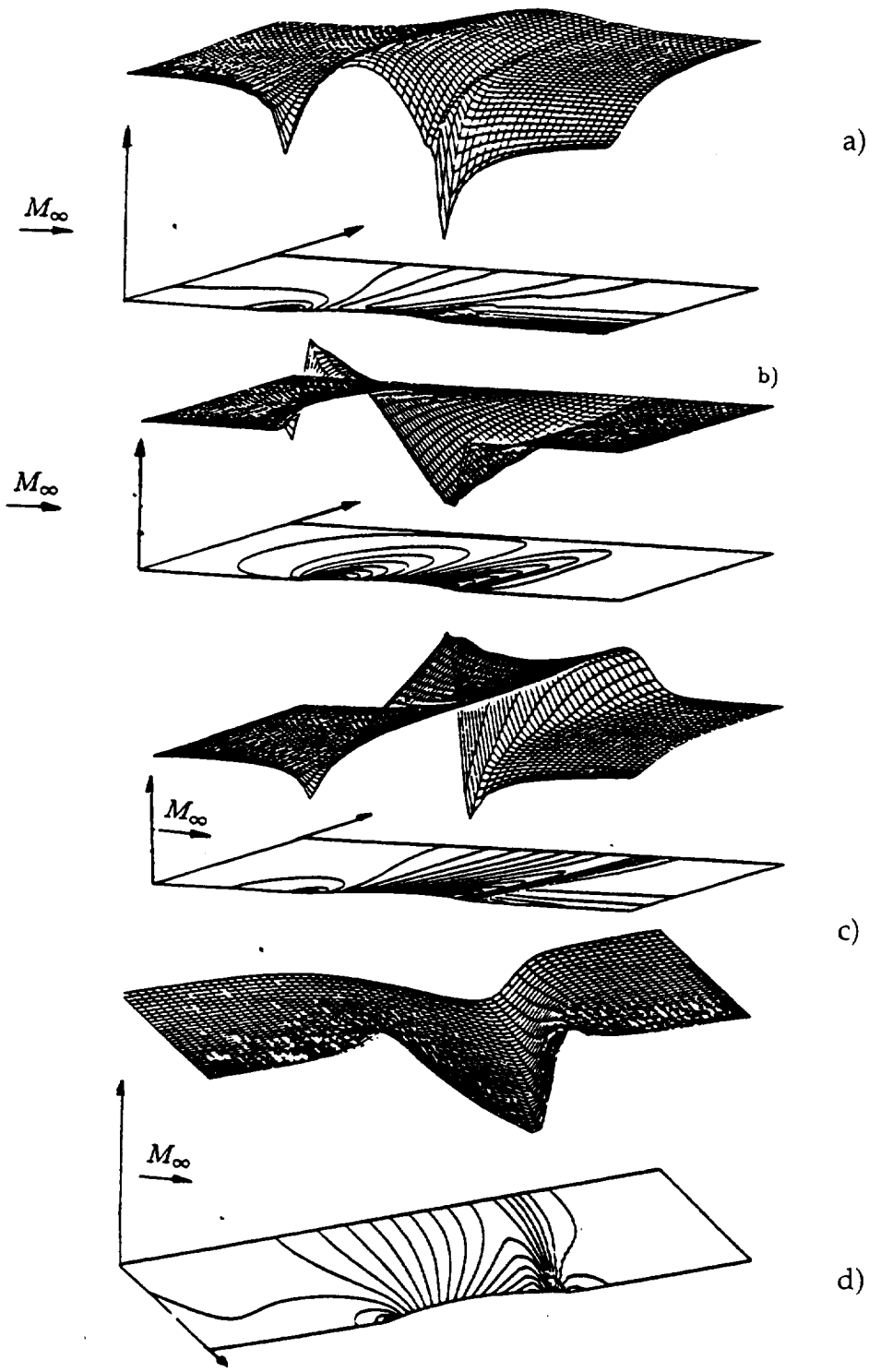


Figure 7.11 *REMI* algorithm TWS^h Euler solution, steady-state, 15% parabolic arc, scalar β=0.2 $\{1\}^T$ a) 65x35 mesh, b) velocity vector field, c) entropy, d) Mach number, e) axial momentum.



igure 7.12 *REMI* algorithm TWS^h Euler solution, steady-state, 15% parabolic arc, Ma_{in} = 0.68, scalar β=0.2 $\{1\}^T$, perspective and contour graphs of, a) axial momentum, b) transverse momentum, c) Mach number, d) pressure.

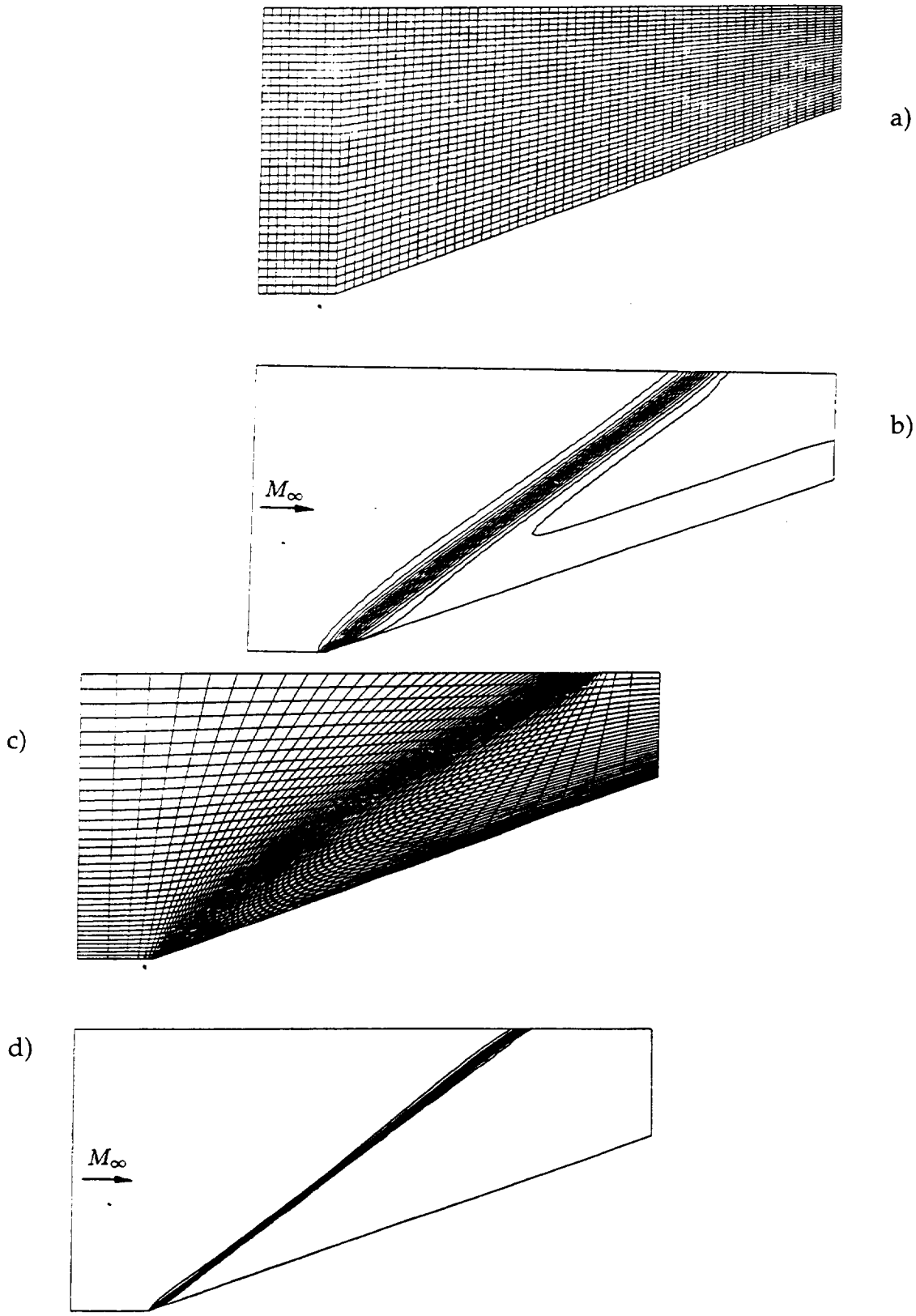


Figure 7.13 *REMI* algorithm TWS^h Euler solution, supersonic wedge flow, $\theta = 20^o$, $\beta = 0.3$ $\{1\}^T$, a) initial 65x35 uniform mesh, b) density isoclines, c) 1st adapted mesh, d) resultant density isoclines.

block-constructed from the base 2x2 macro mesh, each computational mesh contained 65x35 elements. The IRK ODE algorithm was used throughout in non-iterative mode. For the initial sheared uniform body-fitted mesh, Fig. 7.13a), the Euler solution density distribution obtained after 20 time steps, with $CFL_{max} \sim 10$ is graphed in Fig. 7.13b). This *REMI* solution is monotone but only "resembles" the analytical steady-state due to gross diffusion for scalar TWS^h β=0.3. These *REMI* data were interpolated to the solution-adapted mesh with shock region refinement, Fig. 7.13c). Following 40 additional time steps with CFL_{max}~100 the *REMI* density field resolution is much sharper, Figure 7.13d). This isodensity graph is devoid of isolated closed contours, a firm indication of a quality ENO solution.

Figure 7.14a) graphs the second solution-adapted mesh, with maximal clustering along the shock and wedge, yielding very large mesh aspect ratio distortions. The resultant TWS^h-produced ODE system becomes extremely stiff, yet the IRK algorithm converged to a steady-state in six more time steps to $|\delta q|_{max} < 10^{-6}$. CFL_{max} increased to 150 on this mesh, and the resultant density distribution confirms a crisp ENO shock simulation, Fig. 7.14b). Perspective graphs of Mach number and pressure with contours confirm the captured shock quality, Fig. 7.14c)-d). The arrow with label M_∞ shows onset flow direction, and the discrete discontinuity is correctly a straight line (plane) inclined at the analytical solution angle to three significant digits. The shock is absolutely free of precursor under-or over-shoots, and the overall TWS^h solution field is oscillation-free (ENO), including the stagnation streamline and where tangency accrues via weak enforcement (only). Clearly, use of highly non-orthogonal distorted FE meshings is fully admissible for Euler solutions within the TWS^h algorithm with TP quasi-Newton jacobian constructions. Further, block solution-adaptation is verified as a viable methodology, eligible for automation with appropriate error detection schemes.

An extension on this validation, leading to a viscous verification, is supersonic flow in a converging duct formed by an inclined "splitter plate" above a horizontal surface. For onset flow at Ma_∞=2.15, and splitter plate angled at α=-4°, the entire flow remains supersonic, hence BC-IC procedures are unchanged from the previous example. Figure 7.15 summarizes the *REMI* algorithm solution-adaptive re-meshing sequence leading to sharp shock prediction at the plate, nominal resolution of shock reflection off the floor, and undistorted passage of the reflected shock out the upper boundary.

These results, obtained with the scalar β TWS^h formulation in FEMNAS, are ENO. The sole detraction is apparent longitudinal false diffusion in the shock-floor impingement region. A local mesh attraction would correct this, however FEMNAS

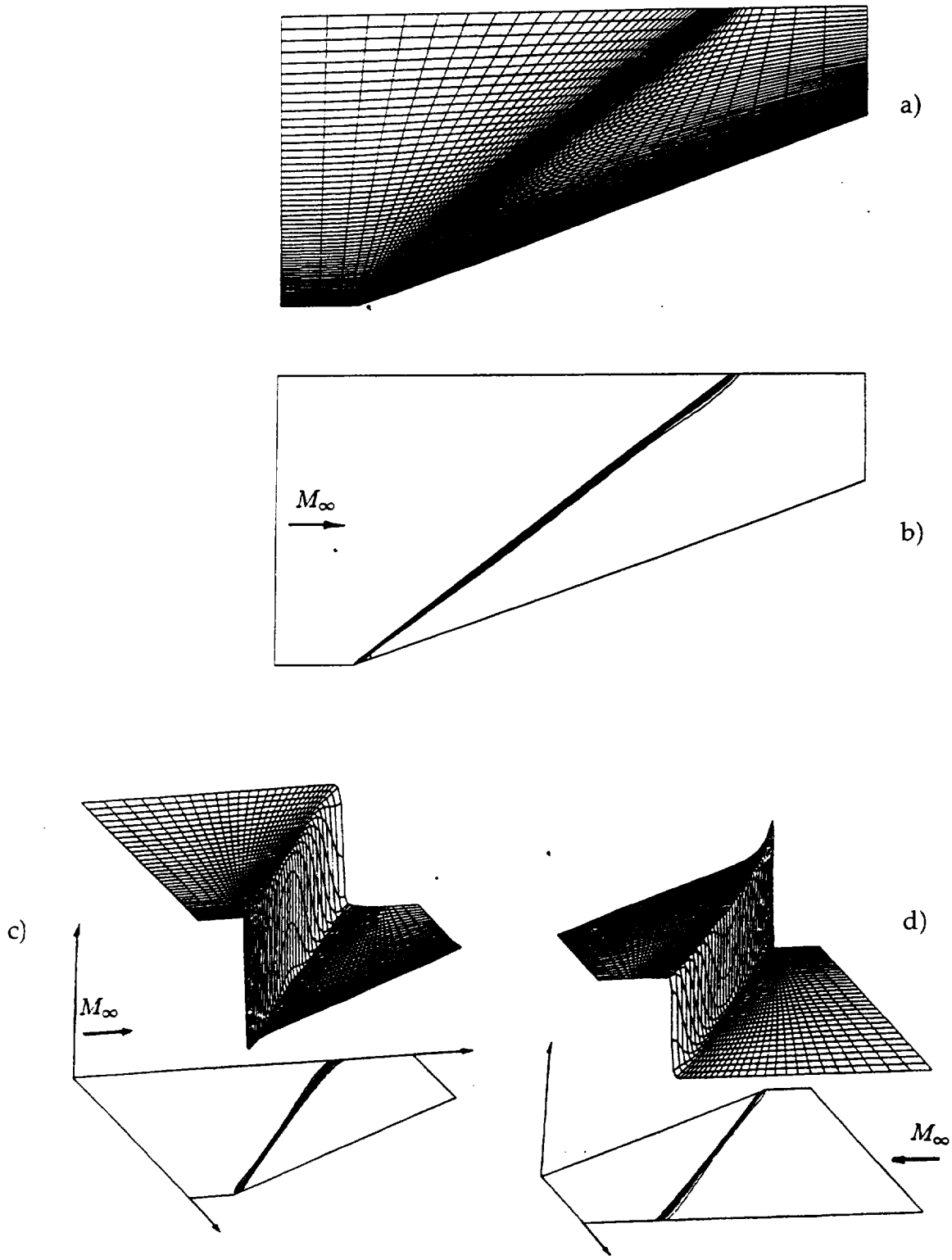


Figure 7.14 *REMI* algorithm TWS^h Euler solution, supersonic wedge flow, θ=20°, β = 0.3 $\{1\}^T$, a) final adapted mesh, b) density isoclines; contour and perspective graphs, c) Mach number, d) pressure.

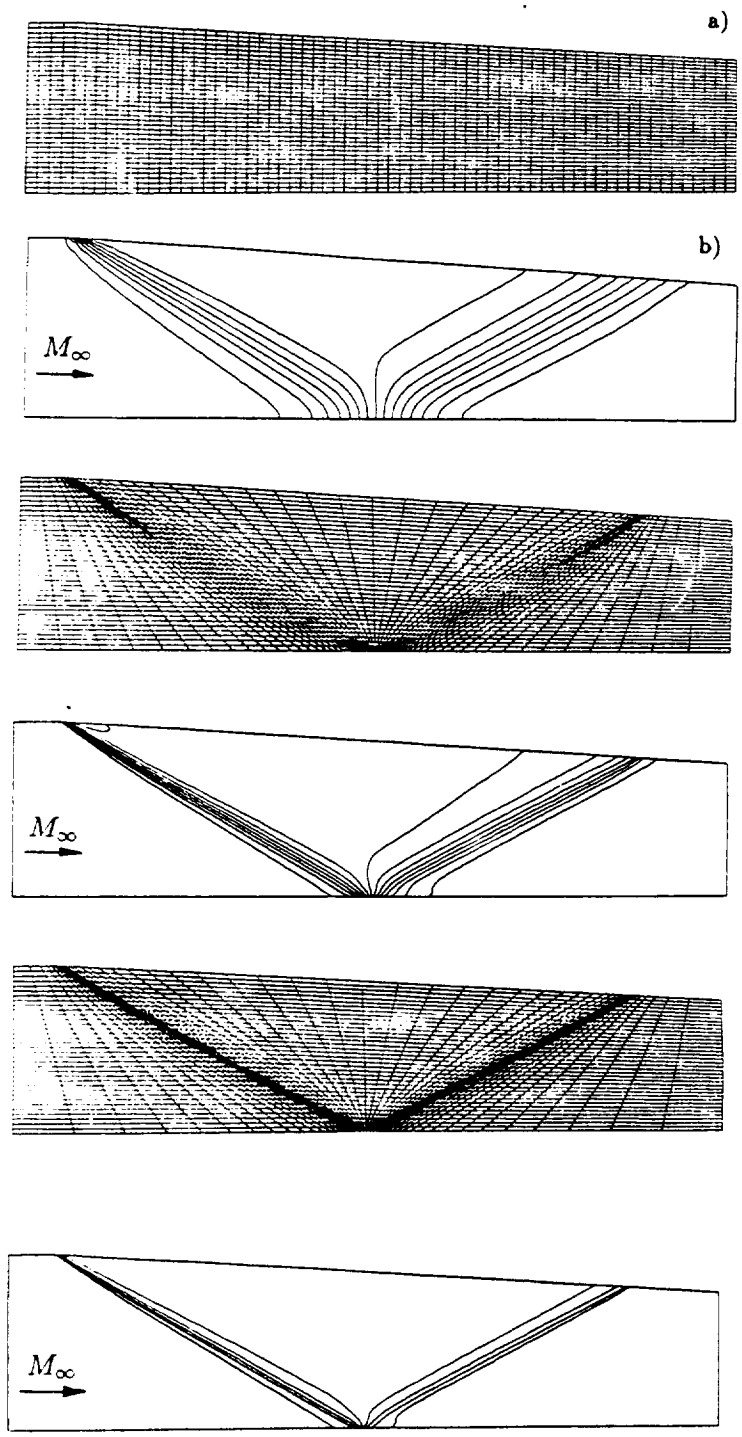


Figure 7.15 *REMI* algorithm TWS^h Euler solution, supersonic shock reflection, β = 0.3 $\{1\}^T$, solution-adapted 65x35 meshing, resultant density isocline distributions.

did not possess this ability in its mesh generator. Figure 7.16 presents contour and perspective surface distributions of pressure and entropy for this *REMI* algorithm solution on the final mesh. A very good approximation to the conical flow analytical solution is exhibited, and the entropy graph confirms sharp creation fronts at both shocks, and a wall layer region downstream of the shock reflection.

7.5 Hypersonic Euler verification, validation, d=2 axisymmetric

The *REMI* FE TWS^h algorithm is verified for inviscid hypersonic flow over a spherical forebody, Iannelli (1991, Ch. 10), using FEMNAS. Figure 7.17a) presents the final solution-adapted 65x35 quad meshing for the blunt body flow at Ma_∞=6.5, and the corresponding steady-state density isocline distribution is ENO, Fig. 7.17b). The transverse (body-normal) scale of these plots is highly stretched for clarity, hence the shock appears much more diffused that in actuality.

Attaching a $7^o/10^o$ bicone to the hemispherical nose, and for Ma_∞ = 8.0, Fig. 7.17c)-d) present adapted-mesh contour and perspective distributions for *REMI* Mach number and density. The scalar TWS^h dissipation parameter is β_q =0.4{1}^{T}, which yields ENO solutions with sharp resolution of shock and forebody flow detail. In Fig. 7.17c), the letters denote stagnation point (C), sphere-cone juncture (D) and bicone juncture (E) in the perspective view. (Again, the transverse plot coordinate is highly stretched for viewing clarity.) With the IRK algorithm, the solution process was stable to CFL_{max} ~ 200, and predicted steady-state bow shock standoff of δ/R=0.135 agrees within 4% with available data, Anderson(1989), where R is hemisphere radius.

This Ma_∞=8 test case affords a benchmark for validating the developed five-species equilibrium reacting air thermodynamics model, Section 2. For the typical wind tunnel attainable stagnation state, no appreciable real-gas effects are generated for this problem. Conversely, for the representative at-altitude condition of p_∞ =0.03 atm and T_∞=221°K, the bow shock temperature rise will induce substantial real gas effects. The comparison inviscid real gas simulation was also stable for CFL_{max} ≈ 200. Real gas effects lead to a decrease in normalized bow shock standoff to δ/R=0.11, which is 18% smaller than the ideal gas prediction. This is expected since the real gas temperature increase across the bow shock is substantially smaller. Figure 7.18 compares stagnation streamline temperature distributions for the real-gas and ideal-gas Euler simulations. The relative shock temperature decrease is accountable to energy absorbed by dissociation and vibrational modes, and eventual formation of nitric oxide and atomic oxygen, Fig. 7.18b).

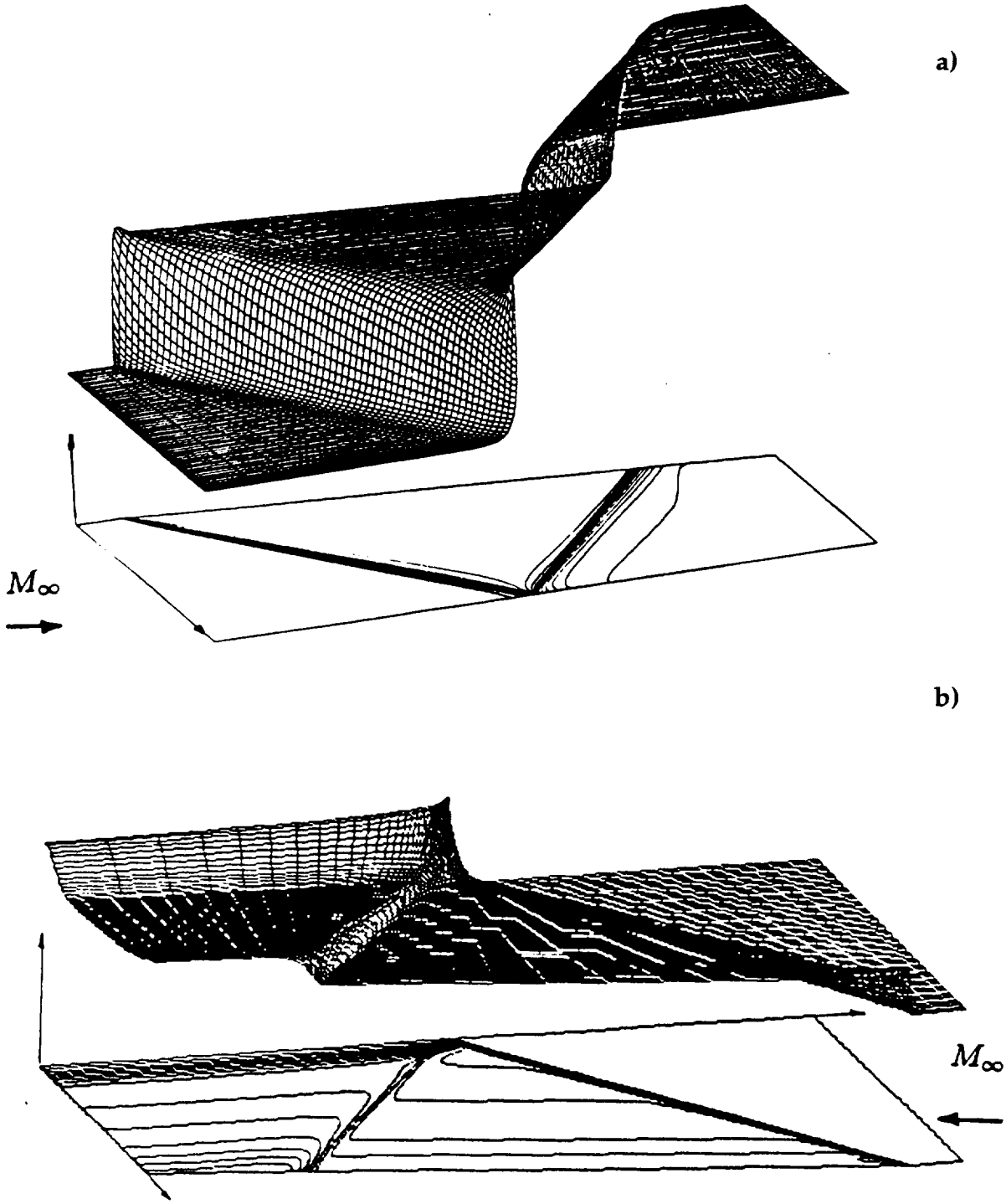


Figure 7.16 *REMI* algorithm TWS^h Euler solutions, supersonic shock reflection, $\beta_a = 0.3\ \{1\}^T$, final mesh, contour and perspective graphs, a) pressure, b) entropy.

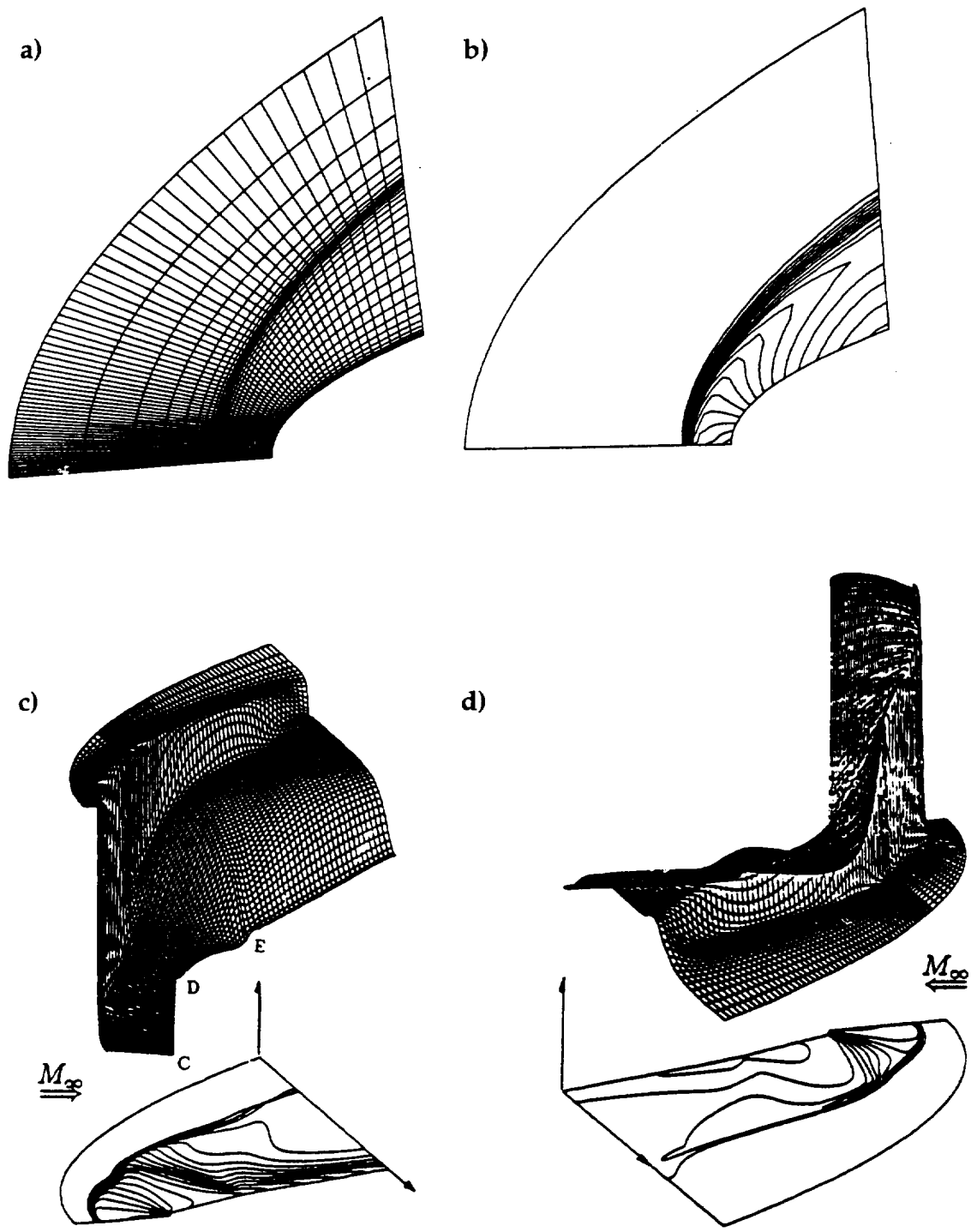
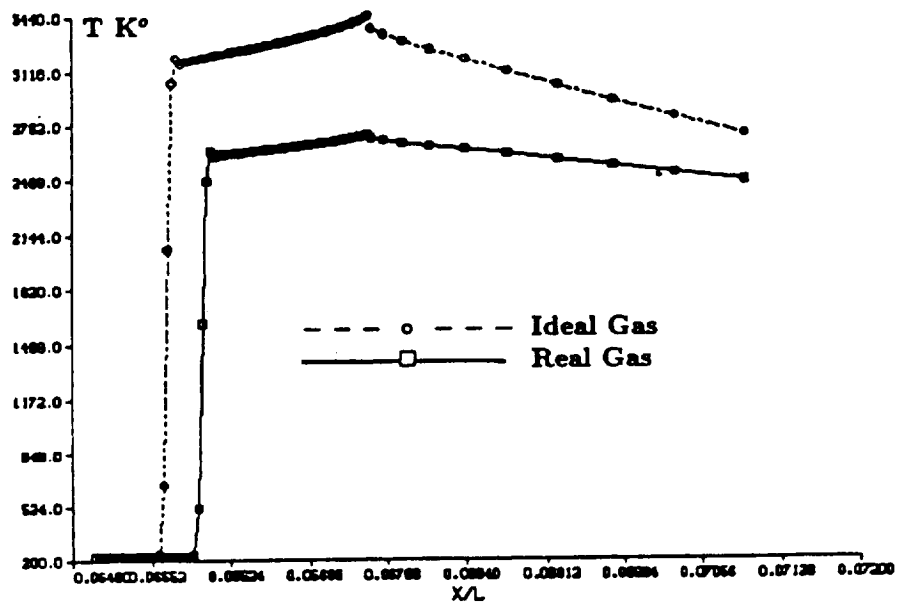
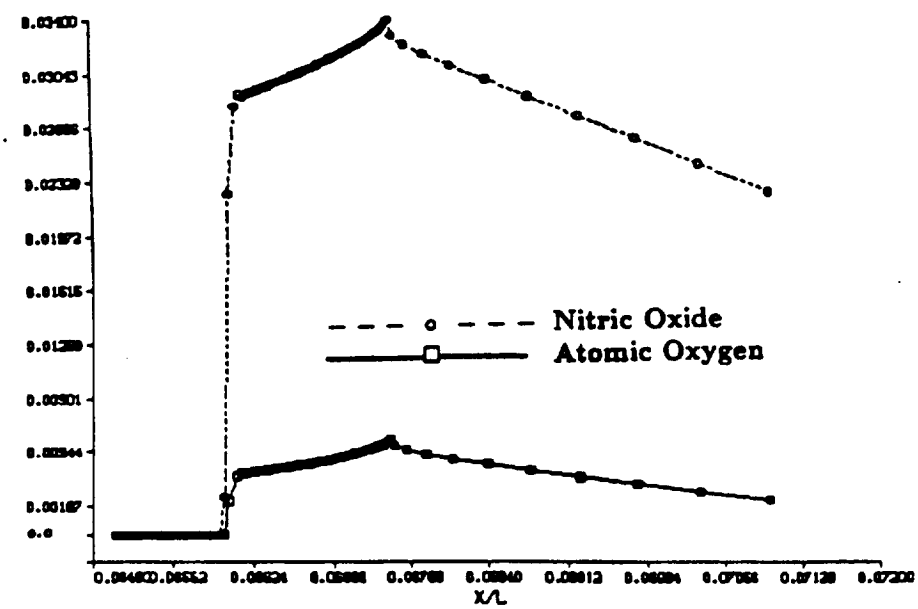


Figure 7.17 *REMI* algorithm TWS^h steady state Euler solutions, hypersonic blunt-body flow adapted 65x35 quad meshes, Ma_∞=6.5, a) mesh, b) density distribution; Ma_∞ = 8.0, contour and perspective surface distributions of c) Mach number, d) density.



a)



b)

Figure 7.18 *REMI* algorithm TWS^h Euler solution, steady-state, Ma_∞=8,

a) ideal-air and real-air stagnation streamline/body surface distributions of temperature,

b) companion real-air species mass fractions

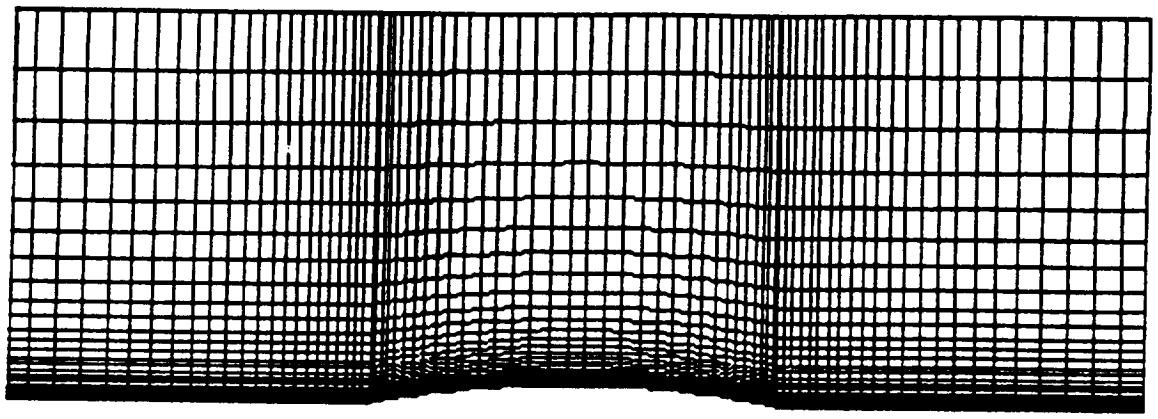
7.6 Viscous transonic benchmark, validation, d=2

The goal is to validate *REMI* Navier-Stokes algorithm (NS) solutions at practical aerodynamics Reynolds numbers. The associated requirement is use of highly refined meshings adjacent to surfaces, which yields very large mesh element aspect ratios, hence ODE system stiffness. One NS benchmark test is a laminar viscous simulation of the parabolic arc Euler benchmark geometry, Fig. 7.11a), with the thickness reduced to 4% such that the Euler solution is shock-free. The resulting FEMNAS 65x45 sheared cartesian meshing is graphed in Fig. 7.19a), which is a 3-block macro construction with non-uniformity generated by geometric progression.

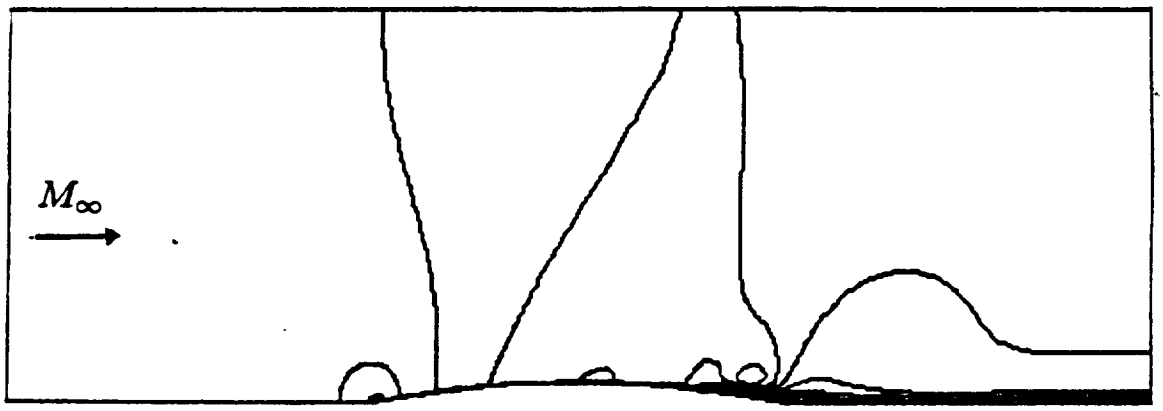
For Reynolds number Re=4.0×10^6, the progression factor was set such that element vertical span adjacent to the airfoil surface was order Re^{-1}. The intention was that TWS^h β-dissipation not dominate near-wall phenomena, recall the stability analysis (5.48)-(4.51). The exterior flow remains subsonic, hence inflow Dirichlet BC are applied to ρ, m_2, and E, pressure is fixed on the outflow plane, and the top boundary is flow tangency with vanishing normal derivatives for all variables. Flow tangency with vanishing normal derivative is applied to the bottom boundary everywhere except on the airfoil surface, where $m_1 \equiv 0 \equiv m_2$ for no-slip. The IC was interpolated quasi-one dimensional Euler analytical solution, follow by switch to no-slip at Re=10^3, hence Reynolds number continuation from Re=10^3 to 4.0×10^6.

Figure 7.19b graphs the principal momentum solution established following ~800 time steps at Re=4.0×10^6 using the IRK algorithm in non-iterative mode. The scalar TWS^h dissipation level was uniform at $\beta_q=0.2\{1\}^T$, $CFL_{max} \approx 100$ was achieved, the L_2 norm of the algorithm residual ranged $10^{-5} < |RQ|_{max} < 10^{-6}$, and the resultant inlet flow Mach number converged to Ma_{in}=0.7. Figure 7.19b) confirms that essentially all computed action lies in the boundary layer, which appears thereon as a "thick line" with bumps. Presenting these data in nodal space greatly stretches the lateral coordinate, Fig. 7.19c), which confirms that there is a complicated double shock-boundary layer interaction on the last third of the airfoil, and upstream of this the laminar boundary layer grows smoothly and modestly.

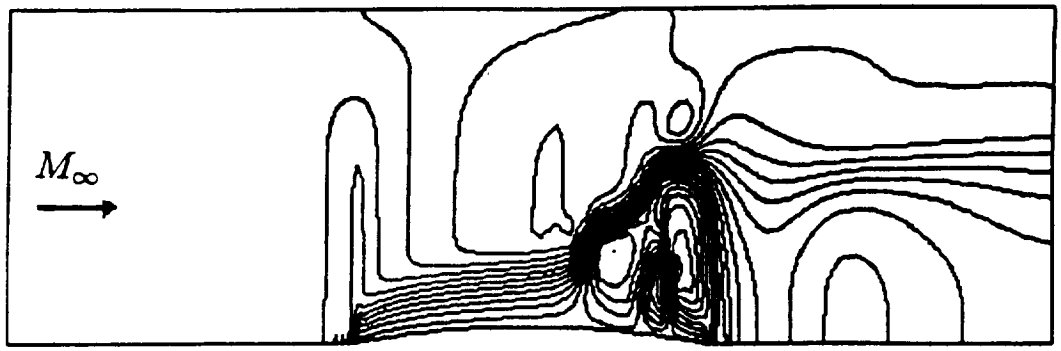
A perspective surface presentation of this solution adds to qualitative assessment, Figure 7.20a), which clearly shows quality boundary layer resolution achieved and a double separation region following 2/3-chord. The solution flow approach to the leading edge stagnation point is ENO, as is the solution exiting all far-field boundaries. Figure 7.20b) presents in perspective the pressure distribution, which clearly shows the stagnation peak and the adverse gradient leading to the first shock and its rebound.



a)



b)



c)

Fig. 7.19 *REMI* algorithm TWS^h Navier-Stokes, laminar, viscous, 4% parabolic arc, Re=4.0x10^6, β=0.2, $\{1\}^T$, a) non-uniform mesh, b) axial momentum, c) axial momentum plotted in nodal space.

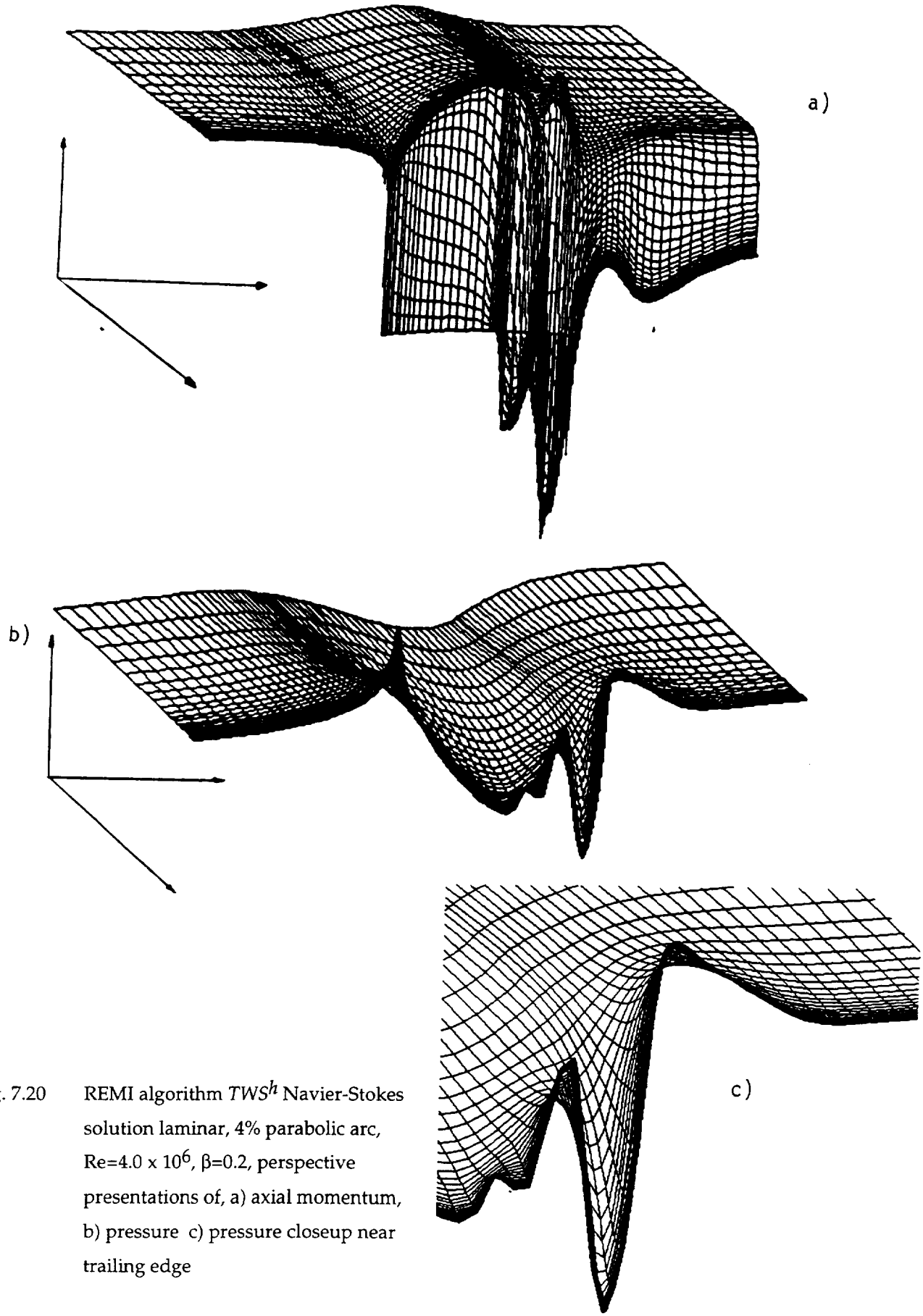


Fig. 7.20 REMI algorithm TWS^h Navier-Stokes solution laminar, 4% parabolic arc, Re=4.0 x 10^6, β=0.2, perspective presentations of, a) axial momentum, b) pressure c) pressure closeup near trailing edge

Figure 7.20c) is a trailing edge region zoom of pressure confirming an ENO solution.

This practical Reynolds number simulation verifies *REMI* algorithm ability to handle high aspect ratio discretizations, and has produced an interesting shock-boundary layer flow interaction embedded strictly within the boundary layer. No independent benchmark or validation data are available, however, even though this makes a geometrically simple candidate for code validation. Therefore, a laminar flow validation was selected corresponding to external shock impingement on a developing flat plate boundary layer in a uniform supersonic free stream.

The problem geometry is that discussed in Fig. 7.15, and Fig. 7.21 graphs the essence of the problem, Degrez, et al (1987). An oblique shock generated by a splitter plate causes boundary layer separation. For inlet flow at Ma_∞ =2.15, the separation and reattachment flow remains laminar for Re_S=0.96x10^5 based on x_S, the shock intersection distance from the plate leading edge. The free stream stagnation state was p_0 = 0.1 atm and T_0=295°K. The BC definitions accordingly are all Dirichlet at inlet, pressure is fixed at outlet only in the subsonic boundary layer region, and tangency with vanishing normal derivative occurs elsewhere, except m_1 =0=m_2 on the plate surface (for no-slip). The IC was an Euler solution, generated as discussed for Fig. 7.15, but augmented with an inviscid stagnation point (single node) at the plate leading edge to promote the corresponding shock. The 65x76 mesh was modestly shock adapted for adequate wall region resolution, *i.e.*, the vertical element span was order Re^{-1}.

The laminar viscous solution evolved smoothly from the quasi-Euler IC using FEMNAS with IRK in the non-iterative mode, and achieved nominal steady state in 400 time steps with $CFL_{max} \approx 200$. Fig. 7.22 presents perspective and contour distributions

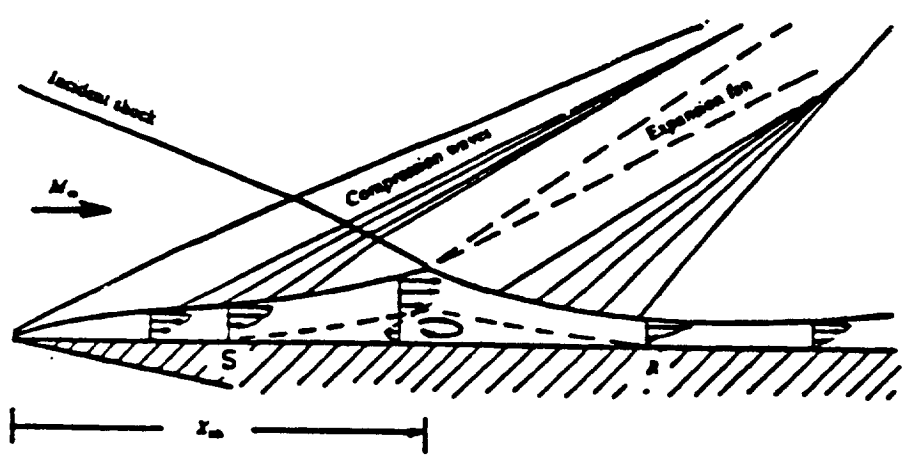


Figure 7.21 Shock-laminar boundary layer validation problem

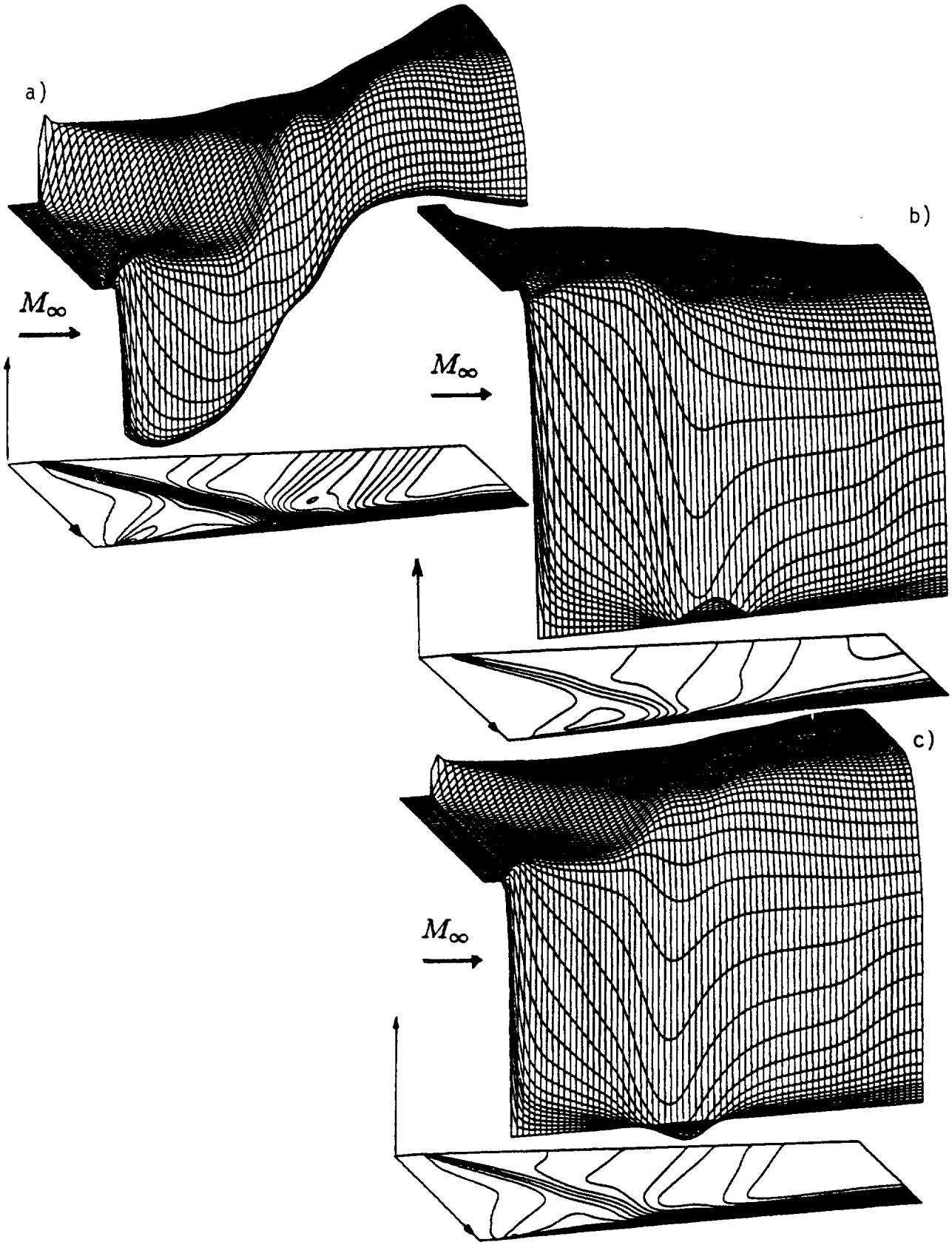


Fig 7.22 REMI algorithm TWS^h Navier-Stokes solution, laminar shock boundary layer, Re=105, β=0.3 $\{1\}^T$, perspective and contour graphs of, a) density, b) Mach number, c) axial momentum.

of *REMI* solution axial momentum, density and Mach number, which confirm attainment of an ENO solution with no spurious boundary reflections. The momentum graph clearly defines the boundary layer separation region. Figure 7.23 summarizes the *REMI* algorithm velocity resolution of this region, and documents the level agreement achieved with experimental data for surface pressure and skin friction distributions, Degrez, et al (1987). Excellent quantitative agreement exists in approach to the separation and within, but the reattachment region shows relatively poorer agreement. The *REMI* simulation predicts a lengthier adverse pressure region, hence the computational reattachment occurred downstream of the experimental data. This could result from secondary shock reflection effects off the top splitter plate, in the experiment, or from local viscous region mesh distortion, with use of averaged metric data (recall section 7.2 discussion). This FEMNAS limitation could aggravate local excess numerical diffusion effects, as was observed in the Euler shock reflection benchmark.

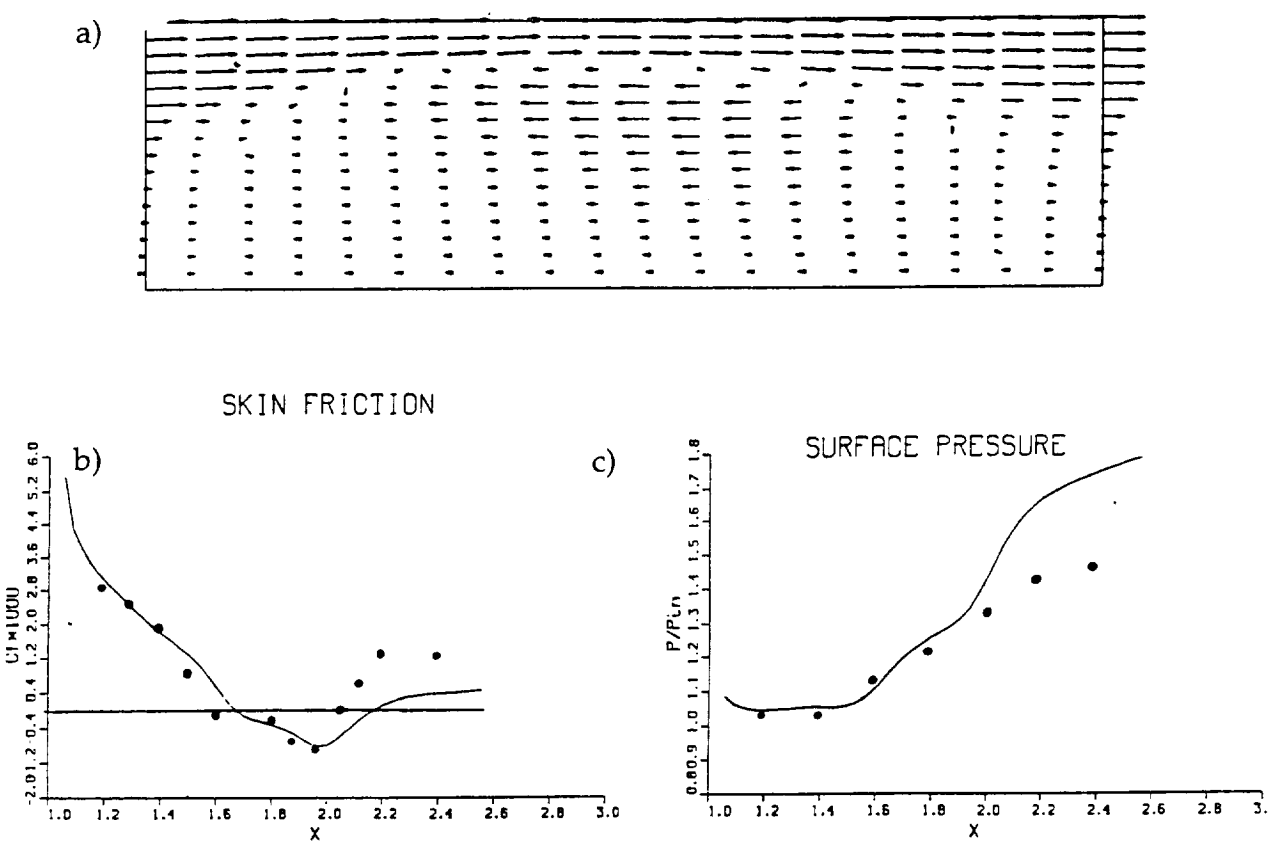


Figure 7.23 Supersonic shock-boundary layer interaction, $Ma_{in} = 2.15$, $Re=10^5$,
a) *REMI* separation region velocity resolution; comparisons on
b) surface pressure c) skin friction, symbols are data from Degrez, *et al* (1987)

8. SUMMARY AND CONCLUSIONS

An "arbitrary grid" CFD algorithm for unsteady aerodynamics applications has been derived, verified and benchmarked for selected Euler and Navier-Stokes descriptions. The theory employs a finite element (FE) spatial semi-discretization of a Taylor series-augmented Galerkin weak statement (WS), which for any FE trial space basis produces a large, coupled non-linear ordinary differential equation (ODE) system. The θ-implicit, single-step Euler family, and a newly derived implicit Runge-Kutta (IRK), algorithm, employ this generated ODE system, producing a large non-linear algebraic matrix equation statement for computing. This matrix solution statement was cast as a Newton algorithm, whereupon a matrix tensor product (TP) factorization was derived as a (hopefully) efficient quasi-Newton iterative approximation replacing sparse matrix solution methods with relatively efficient block-banded, iterative mesh sweeping Gauss elimination processes.

In the execution of this project, and the subsequent longer term development and verification process, close attention was given to key theoretical issues regarding well-posedness, hence boundary conditions, stability and parasitic stiffness resulting from finely graded meshes for viscous simulations at aerodynamic Reynolds numbers. A Lyapanov stability analysis yielded theoretical verification of Navier-Stokes boundary conditions, enforceable via WS-generated surface integrals, appropriate everywhere, including subsonic outflows, involving linear combinations of pressure, surface derivatoric tractions and heat flux vector. The implicit Runge-Kutta ODE algorithm coefficient set was derived to combine classical second-order time accuracy with *B*-consistency and *B*-stability. This combination of desirable features leads to a significant performance improvement over either the backwards Euler ($\theta=1$) or trapezoidal ($\theta=0.5$) ODE algorithms regarding admissible step-size (Courant number), freedom from artificial dissipation ($\theta=1$), and assimilation of parasitic stiffness associated with admissible Navier-Stokes meshings. A key post project contribution was the derivation of consistent metric data handling for TP factorization quasi-Newton jacobians, first in two dimensions and now in three.

The algorithm features discussed above are potentially applicable to selection of *any* FE domain shape, *e.g.*, triangles/tetrahedra and/or quads/hexahedra, and any degree-polynomial basis defined thereon. The triangle/tetrahedron family has become associated with the words "unstructured mesh," while the quad/hex family is usually considered as a "structured mesh." In fact, structured or unstructured meshings can be

PRECEDING PAGE BLANK NOT FILMED

constructed from either element shape, with the quad/hex family exhibiting required versatility via local embeddings and trial space basis enrichment ("p"-elements) with "hanging nodes," i.e., degrees-of-freedom that occur as vertex-mid-edge on adjacent element domains, c.f., Bass, et al (1993). (The triangle/tetrahedron family does not typically require hanging nodes, although they certainly are theoretically admissible.) Further, either family of elements is admissible for use with overset "Chimera" meshes using data handling and interpolation procedures well established in the finite volume CFD community.

As summarized in the Introduction, we made the choice at project outset to seek verification/benchmark/validation results for only the quad/hex element family, and to further restrict consideration to the linear k=1 FE basis form only. This was based on the observation that meshing requirements for genuine Reynolds numbers really precluded use of explicit ODE methods, and that the TP quasi-Newton algorithm showed great promise to maintain algorithm accuracy and efficiency for viscous (and eventually, turbulent) simulations on meshes containing large aspect ratio elements, as would be required to resolve flow details in surface normal directions. Based on this decision, the following conclusions may be drawn from the results of this project (and its extension).

Arbitrary meshing

Generated computational results verify that the quad/hex finite element family in the TWS^h is amenable to substantial non-cartesian distortion while maintaining accuracy for inviscid reaches of the flowfield and convergence for the TP factorization quasi-Newton iteration. These confirming data were generated mostly for d=2 and axisymmetric geometries using the FEMNAS code. The confirming data for d=3 are just now being generated using ***AKCESS*.AERO** in a parallel processing computer environment. There appears little practical restriction on mesh aspect ratios or angularity for shock capturing using efficient metric data averaging techniques. This meshing arbitrariness extends to viscous region simulations *only* when accurate numerical quadrature methods are employed to form dissipative flux vector weak statement contributions.

This point was unknown during the FEMNAS shock-boundary layer validation test. The NS dissipative operator was formed using metric-averaging on elements that departed significantly from parallelograms and algorithm stability problems were encountered in the shock footprint region on the plate. The TWS^h β-dissipation term would also suffer from this inaccuracy also. However, it was exponentially "shut-off"

in the boundary layer lower reaches, hence did not affect loss of stability in our opinion. The concluding opinion is that any and all dissipation-type forms (second derivatives) in Euler/RaNS systems must be accurately evaluated for arbitrary meshing being an attribute of the quad/hex FE element family in a TWS^h algorithm.

Numerical dissipation

The TWS^N theory provides an in-the-continuum framework for deriving tensor-invariant augmentations to the RaNS system for numerical dissipation. Work funded in this original contract (Baker and Kim, 1987) verified a wide range of current practice dissipative algorithms belong to the theory. Work in progress now (Iannelli, 1993, 1994) confirms that this theory extends to encompass many flux vector splitting CFD methods recognized today as TVD/ENO finite volume methods.

Ultimately, robustness of the artificial dissipation procedure is key to success for shock capturing aerodynamics applications. A significant simplification to the TWS^N theory-derived β dissipation term was made to establish the scalar β-form used in this project. With accurate metric data handling and block mesh adaptation, this construction is verified to admit prediction of monotone shocks for the range transonic to hypersonic, oblique and normal. The surface perspective data presentations are particularly graphic for this verification, confirming fully the shock quality attainable via the k=1 FE basis implementation of TWS^h. However, while not fully confirmed, evidence exists that excessive "crosswind diffusion" accrues to the scalar construction which accounts for the move (in ***AKCESS*.AERO**) to the tensor form with options to guarantee (the need for) uniform positivity. Further, the move to a numerical quadrature for the β-terms, as well as the genuine viscous/heat conducting diffusion terms, could well lead to improved performance for either formulation, or any successors, on arbitrary meshes.

Tensor product matrix iteration

As stated in the Introduction, and fully detailed in Section 6, the TP quasi-Newton iteration algorithm requires very careful attention to yield an accurate formulation, principally metric data handling. The theory emerged to accomplish this (in two dimensions) in the two years of project extension, following expiration of the original project performance period. Now with ***AKCESS*.AERO**, the three dimensional form is approaching operational verification. Certainly alternatives exist, such as line Gauss Siedel, however the TP formulation appears particularly well suited to a parallel processing implementation on emerging machine architectures.

The generated verification and benchmark data confirm that, correctly formed, the TP matrix iteration algorithm is convergent and stable on non-cartesian meshes for solutions in inviscid flow reaches. Based on this assessment, the experienced stability problems in the shock boundary layer interaction validation test are probably more associated with matrix formation inadequacies than in the linear algebra construction. The *template* input procedures now available will certainly expedite the range of computational experiments required to determine existence of the optimal form.

Implicit ODE methods

One precept of this project was that genuinely implicit ODE methods must be used in distinction to the explicit or operator splitting methods more familiar to the FE unstructured mesh CFD aerodynamics research community. The TP matrix iteration was a key ingredient to this goal and the emergence of the implicit-Runge-Kutta algorithm was the complement. In non-iterative mode, it is second order accurate, stiffly stable and amenable to use with large CFL number on highly distorted meshes. While requiring twice the work of the θ-implicit family, firm evidence is established that its use may be more efficient than the equivalent θ-delta form. It certainly adds to the efficiency of a NS simulation, in comparison to alternative unstructured mesh algorithms that are not as well suited to implicit ODE algorithms.

Algebraic block macro meshing

Algebraic block mesh constructions are particularly well suited to the developed TWS^h FE algorithm, since no verified need exists to maintain a degree of mesh regularity in either inviscid or viscous flow reaches. Therefore, as an alternative to on-the-fly mesh enrichment/coarsening of shock tracking unstructured algorithms, block mesh adaptation for shock layer refinement may be a computationally attractive alternative. The underlying strategy for either approach is degree of freedom economy, usually at the expense of memory, using error detection/estimation algorithms based on equipartition of estimated error density. These error estimation theories are now well established; coupling the verified TWS^h algorithm mesh arbitrariness versatility to an estimator would appear the logical replacement to the manual mode employed for benchmarking.

In summary, this project has succeeded to a degree considerably smaller than originally anticipated, due principally to severe underestimation of the difficulties associated with converting a cleanly derived theory into code practice. During the original two year performance period we succeeded in establishing only a verification

level TP factorization procedure, accurate on nominally cartesian meshes. With CMC support, Iannelli started his dissertation project near the end of this period, hence derived the incisive analyses required to move to adaptive block meshing with quality ENO shock capturing and accurate TP constructions. His two dimensional FEMNAS code was developed in the process, hence served as the research test bed to validate the determined set of algorithm design decisions.

The theory and code practice has now been pushed to three dimensions via the *AKCESS.** software platform which provides a genuinely versatile, externally programmable venue to validate ideas and theory variations as developed. Certainly, as painfully learned in this project, an "Achilles heel" of CFD is code inflexibility, an unreliability, hence the need for constant reprogramming for every step forward. Hopefully, the emergence of *AKCESS.** as a project consequence, in the longer term will warrant the financial and personnel efforts expended in its creation.

*AKCESS.** is in use today supporting research on a new *p*-element embedding strategy, in the base k=1 weak statement formulation, for monotonicity control in shocks and wall layer regions, without requiring extensive mesh adaptation or artificial diffusion, (Roy and Baker, 1993). For example, Fig. 8.1 illustrates results comparing standard vs. *p*-embedded solutions for the (inviscid) minimum resolution (2-node) traveling square wave, and the steady non-linear viscous Burgers problem (Re=10^5) emulating a shock in viscous flow. The classical k=1 and/or k=2 solutions are grossly distorted by dispersion error, while nodally-exact ENO solutions on *coarse* meshes, with *no* added artificial diffusion, accrue to *p*-embedding on the linear basis algorithm.

This newest FE methodology is being "immediately" assimilated into *AKCESS.** for compressible and incompressible Navier-Stokes benchmarks and verifications. Thereby, the new CFD technology base evolves in the "production code," hence little time is wasted in transition from a research code. The lesson from this project is well learned, and the end result is rapidly bearing fruit.

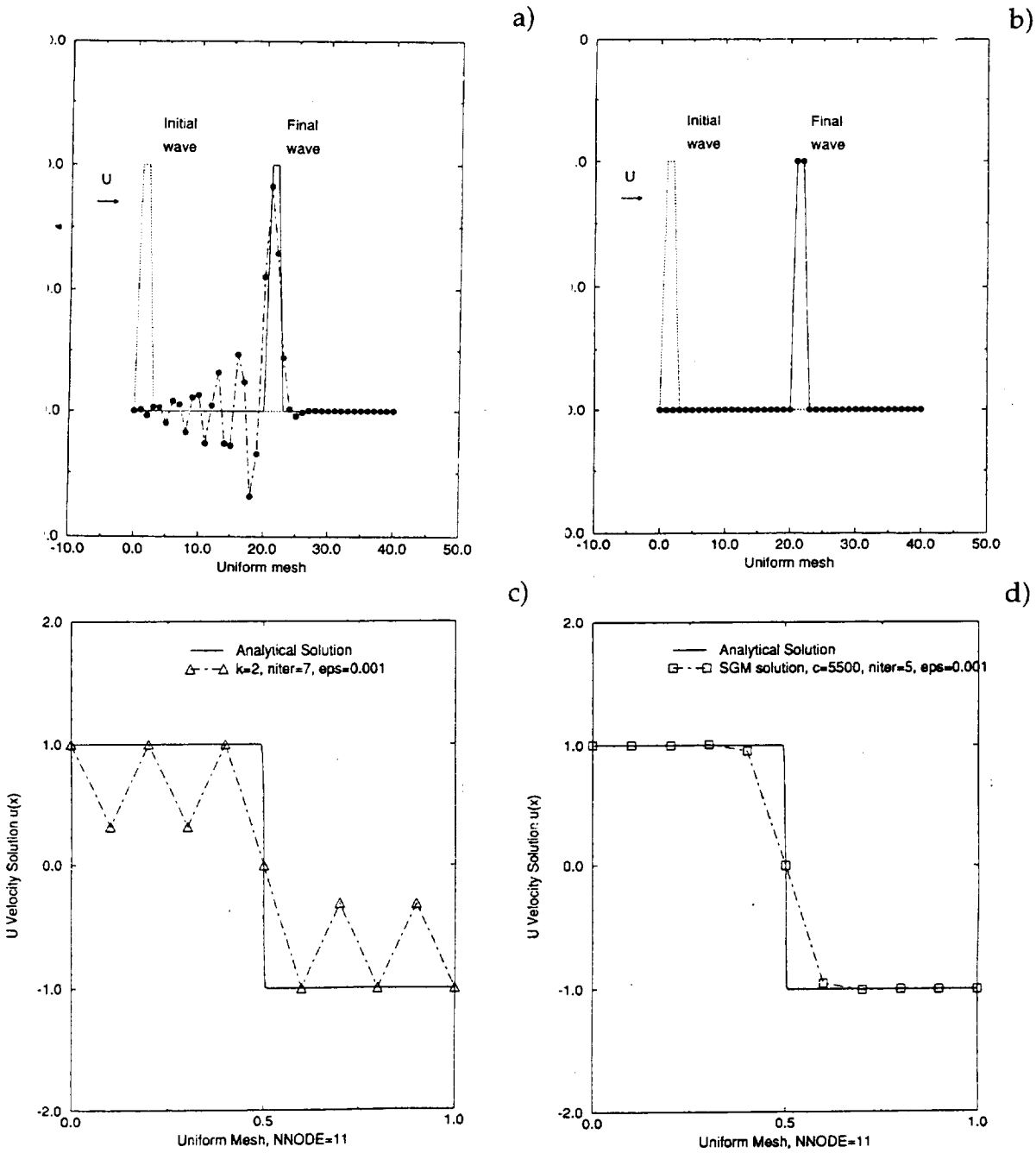


Figure 8.1 Sub-grid p-embedding FE verification problems, inviscid square wave, a) standard WS^h solution, k=1 or 2, b) p-embedded solution, k=1; viscous Burgers shock simulation, Re=10^5, c) standard WS^h, k=1 or 2, solution, converged to 10^3 d) p-embedded solution, k=1, converged to 10^{-9}.

References

Anderson, J. D. Jr. (1982), *Modern Compressible Flow with Historical Perspective*, McGraw-Hill, New York.

Anderson, J. D. Jr. (1989), *Hypersonic and High Temperature Gas Dynamics*, McGraw-Hill, New York.

Baker, A.J. & Pepper, D.W. (1991), *Finite Elements 1-2-3*. McGraw-Hill, New York .

Baker, A. J. & Roy, S. (1993), A *p*-Element Sub-grid Embedding for Stability. Manuscript submitted to Journal of Comp. Physics New York.

Baker, A. J. & Kim J. W. (1987), A Taylor Weak Statement Algorithm for Hyperbolic Conservation Laws. Int. Journal Num. Mtd. Fluids, p. 7, 489-520, New York.

Baker, A. J. & Chaffin, D. J.(1994), Analysis on the Taylor Weak Statement for Dispersion Error Control. Manuscript, UT.K.

Baker, A. J. (1983), *Finite Element Computational Fluid Mechanics*. Hemisphere Pub. Corp. New York.

Baldwin, B.S. & Lomax H. (1975), Thin Layer Approximation and Algebraic Model for Separated Turbulent flow. AIAA 16th Aerospace Center meeting, AIAA-78-0257.

Barth, T. J.(1991), Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes.

Bass, J. *et al* (1993), A New Approach for Solving Navier-Stokes Equations on Unstructured Grids Based on Adaptive Methods and Operator Splitting. Report TR.-93-07, Comp. Mech. Co., Austin, Tx .

Beam, R. M. & Warming, R. F (1976), An implicit finite-difference algorithm for hyperbolic systems in conservation law form. Journal Comp. Physics, p. 87-110

Chaffin, D. J. & Baker, A. J. (1994), On a Taylor Weak Statement Finite Element Methods for Computational Fluid Dynamics. J. Comp. Physics, submitted.

Courant, R. & Hilbert, D. (1929), *Methods of Mathematical Physics*. John Wiley, N. Y.

Degrez, G., Boccadoro, C. H. and Wendt, J. F. (1987), The Interaction of an Oblique Shock Wave with a Laminar Boundary Layer Revisited. An Experimental and Numerical Study. Journal of Fluid Mechanics, New York, p. 177, 243-263 .

Donea, J. (1984), A Taylor-Galerkin Method for Convective Transport Problems. Int. Journal. Numerical. Methods in Engineering. p. 20, 101-119.

Dutt, P. (1988), Stable Boundary Conditions and Difference Schemes for Navier-Stokes Equations. SIAM Journal. Numerical Analysis. p. 25, 245-267.

Freels, J.D. & Baker, A.J. (1990), Symbolic Stability Analysis for space-marching of reduced Reynolds-averaged Navier-Stokes Equation Systems. AIAA 28th Aerospace Science Meeting, AIAA-90-0584.

Gresho, P.M. (1991), Incompressible Fluid Mechanics: Some Fundamental Formulation Issues. Annual Review of Fluid Mechanics. p. 23: 413-453.

Iannelli, G. S. (1991), A Globally Well-posed Accurate and Efficient Finite Element CFD Algorithm for Compressible Aerodynamics. PhD Dissertation, U. T.K.

Iannelli, G. S. (1993), Conservative-Variable Average States for Equilibrium Gas Multi-dimensional Flows. NASA Tech Report ICOMP-92-05.

Johnson, G. M. (1982), Relaxation Solution of the Full Euler Equations. Springer-Verlag, Lecture Notes in Physics, p. 170, 273-279.

Liou, M. S. & vanLeer, B. (1988), Choice of Implicit and Explicit Operators for the Upwind Differencing Method. Tech. Paper AIAA 88-0624, 26th Aerospace Sciences Meeting, Reno, NV.

Loehner, R. Morgan, K., and Zienkiewicz, O. C. (1984). An Adaptive Finite Element Method for High-speed Compressible Flow. Presented at the 9th International Conference on Numerical Methods in Fluid Dynamics. p.7, Institute for Numerical Methods in Engineering. University of Swansea, Wales. AIAA Tech. Library.

Lombard, C. K., Oliger, J.and Yang, J.Y (1982), A Natural Conservative Flux Difference Splitting for the Hyperbolic Systems of Gas dynamics. Tech. Paper AIAA 82-0976. Third Joint Thermophysics, Fluids, Plasma and Heat Transfer Conference, St. Louis, Missouri.

MacCormack, R. W. (1969), The Effect of Viscosity in Hypervelocity Impact Cratering, Tech. paper AIAA-69-354 .

Manning, Ruth Ann, Manhardt, P.D., & Orzechowski, J.A., (1993), Parallel 3-D Finite Element Geologic Basin Modeling Code, SIAM conference on Mathematical and Computational Issues in the Geosciences, Houston, Texas.

Moin, P. (1992), The Computation of Turbulence. *Aerospace America* p. 42-46.

Oden J. T. (1986), Strouboulis, T., & Devloo, P, Adaptive Finite Element Methods for the Analysis of Inviscid Compressible Flow: Part I. Fast Refinement/Unrefinement and Moving Mesh Methods for Unstructured meshes. Computer Methods in Applied Mech. & Engr. p. 59, 327-362.

Oden, J.T. and Reddy, J. N. (1976), *An Introduction to the Mathematical Theory of Finite Elements*. Wiley-Interscience, New York.

Park, C. & Yoon, S. (1989), A Fully Coupled Implicit Method for Thermos-chemical Non-equilibrium Air at Sub-Orbital Flight Speeds.Tech. paper, AIAA 89-1974, 9th Computational Fluid Dynamics Conference, Buffalo, New York.

Pulliam, T. H. (1986), Efficient Solution Methods for the Navier-Stokes Equations, Lecture Notes, Von Karman Institute Lecture Series, Numerical Techniques for Viscous Flow Computation in Turbomachinery Bladings. Brussels, Belgium.

Roe, P. L. (1981), Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes, Journal of Comp. Physics. p. 43, 357-372.

Roy, S. & Baker, A. J. (1993), A Weak Statement Perturbation CFD Algorithm with High-Order Phase Accuracy for Hyperbolic Problems.Journal Comp. Methods.in Applied Mechanics & Engineering. (In review).

Roy, S. and Baker, A. J. (1993), A Non-linear Sub-grid Embedding Weak Statement Algorithm for Optimal Adaptive CFD Applications.Journal. Comp. Physics, (in review).

Schmidt, W. & Jameson, A. (1982), Journal CFD Euler Solutions as Limit of Infinite Reynolds Number for Separation Flows and Flows with Vortices. Springer-Verlag, p. 170, 468-473.

Steger, J. L. & Warming, R. F. (1981), Flux Vector Splitting of the Inviscid Gas dynamics Equations with Applications to Finite Difference Methods. Journal Comp. Physics. p. 40, 263-293.

Strickwerda, J. C. (1977), Initial Boundary Value Problems for Incompletely Parabolic Systems. Pure Applied. Math. p. 30, 797-822.

VanLeer, B. (1982), Flux-Vector Splitting for the Euler Equations. Lecture Notes in Physics, p. 170, 507-512.

Varga, R. S. (1962), *Matrix Iterative Analysis* Prentice Hall, Englewood Cliffs, NJ

Vinokur, M. & Montagne, J.L. (1990), Generalized Flux-Vector Splitting and Roe Average for an Equilibrium Real Gas. Journal of Computational Physics, p. 89, 276-300.

White, F. M. (1978), *Viscous Flow* McGraw-Hill, New York.

Williams, P. T. (1993), A three-dimensional, Time-Accurate Incompressible Navier-Stokes Finite Element CFD Algorithm.PhD dissertation, University of TN.

Yakhot, V. and Orszag, S.A., (1986), Renormalization Group Analysis of Turbulence Basic Theory, Journal Sci. Comp. 1, 3.

APPENDICES

APPENDIX A
AKCESS.AERO *REMI* template, *d*=2, Newton

```
INTEGRATION FACTORS
  INITIAL_TIME
  FINAL_TIME
  PROBLEM_CONVERGENCE_CRITERIA
  MAXIMUM_CHANGE_IN_Q__(DQ)
  INITIAL_TIME_STEP
  TIME_STEP_MULTIPLIER
  MAXIMUM_TIME_STEP
  CRITERIA_TO_RAISE_MAX_TIME_STEP
  MAXIMUM_NUMBER_OF_STEPS
  MAXIMUM_NUMBER_OF_ITERATIONS_PER_STEP
  ITERATION_CONVERGENCE_CRITERIA
  THETA__IMPLICITNESS_FACTOR
  MAXIMUM_VALUE_OF_ANY_DELTA_Q

TRANSFORMATION ARRAYS
    ETKJ    1.
    DETJ    1.
#    DETE    0.

BOUNDARY CONDITIONS
        RHO  ETOT  M1  M2  PRSC     # ORDER
 DIR_RT    D    D   0   0    0     # DIRICHLET
 DIR_M1    0    0   D   0    0     # DIRICHLET
 DIR_M2    0    0   0   D    0     # DIRICHLET
 NO_SLIP   0    0   D   D    0     # NO SLIP WALL
 DIR_PRS   0    0   0   0    D     # DIRICHLET
 THR_IN  [3 -3] [3 -3] [3 -3] [3 -3]   0    # THROUGH FLOW
 THR_OUT  [4 -4] [4 -4] [4 -4] [4 -4]   0    # THROUGH FLOW
 BLANK     D    D   D   D    D   # NO SLIP WALL

TITLE          **** TEMPLATE FILE TEMP.CNS2D.REMI ****
 CNS2D  TWS ALGORITHM,  TENSOR MATRIX A JACOBIANS (12/23/93)

RESIDUALS

 RHO   1   #  VARIABLE, SET NO., --- [M] * {[RHO.NEW] - [RHO.OLD]}
()()()(;1)(B200)(-RHO)
 RHO   2   #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
(-)()(U1+U2)(102;0)(B3010)(RHO)
+(-)()(U1+U2)(304;0)(B3020)(RHO)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(RHO)
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(RHO)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(RHO)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(RHO)
 RHO   3   #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(RHO)
+()()(U1+U2)(304;0)(B3020)(RHO)
+()()(U1+U2)(102;0)(B31P1)(RHO)
+()()(U1+U2)(304;0)(B32P2)(RHO)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(RHO)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(RHO)
```


PRECEDING PAGE BLANK NOT FILMED

```
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(RHO)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(RHO)
 RHO   4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(RHO)
+()()(U1+U2)(304;0)(B3020)(RHO)
+()()(U1+U2)(102;0)(B31P1)(RHO)
+()()(U1+U2)(304;0)(B32P2)(RHO)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(RHO)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(RHO)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(RHO)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(RHO)

 ETOT   1  #  VARIABLE, SET NO., --- [M] * {[ETOT.NEW] - [ETOT.OLD]}
 ()()()(;1)(B200)(-ETOT)
 ETOT   2  #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
 (-)()(U1+U2)(102;0)(B3010)(EP)
+(-)()(U1+U2)(304;0)(B3020)(EP)
+(PDUM2,PEI)()()(1122;-1)(B211)(TEMP)
+(PDUM2,PEI)()()(3344;-1)(B222)(TEMP)
+(PDUM2,PEI)()()(1324;-1)(B221)(TEMP)
+(PDUM2,PEI)()()(1324;-1)(B212)(TEMP)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(ETOT)
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(ETOT)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(ETOT)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(ETOT)
 ETOT   3  #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(EP)
+()()(U1+U2)(304;0)(B3020)(EP)
+()()(U1+U2)(102;0)(B31P1)(EP)
+()()(U1+U2)(304;0)(B32P2)(EP)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(ETOT)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(ETOT)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(ETOT)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(ETOT)
 ETOT   4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(EP)
+()()(U1+U2)(304;0)(B3020)(EP)
+()()(U1+U2)(102;0)(B31P1)(EP)
+()()(U1+U2)(304;0)(B32P2)(EP)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(ETOT)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(ETOT)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(ETOT)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(ETOT)

 M1    1  #  VARIABLE, SET NO., --- [M] * {[M1.NEW] - [M1.OLD]}
 ()()()(;1)(B200)(-M1)
 M1    2  #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
 (-)()(U1+U2)(102;0)(B3010)(M1)
+(-)()(U1+U2)(304;0)(B3020)(M1)
+(EULER)()()(1;0)(B201)(PRSC)
+(EULER)()()(3;0)(B202)(PRSC)
+(PDUM2,REI)()()(1122;-1)(B211)(U1)
+(PDUM2,REI)()()(3344;-1)(B222)(U1)
+(PDUM2,REI)()()(1324;-1)(B221)(U1)
+(PDUM2,REI)()()(1324;-1)(B212)(U1)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(M1)
```

```
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(M1)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(M1)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(M1)
 M1    3  #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
+()()(U1+U2)(102;0)(B3010)(M1)
+()()(U1+U2)(304;0)(B3020)(M1)
+()()(U1+U2)(102;0)(B31P1)(M1)
+()()(U1+U2)(304;0)(B32P2)(M1)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(M1)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(M1)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(M1)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(M1)
 M1    4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(M1)
+()()(U1+U2)(304;0)(B3020)(M1)
+()()(U1+U2)(102;0)(B31P1)(M1)
+()()(U1+U2)(304;0)(B32P2)(M1)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(M1)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(M1)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(M1)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(M1)

 M2    1  #  VARIABLE, SET NO., --- [M] * {[M2.NEW] - [M2.OLD]}
 ()()()(;1)(B200)(-M2)
 M2    2  #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
 (-)()(U1+U2)(102;0)(B3010)(M2)
+(-)()(U1+U2)(304;0)(B3020)(M2)
+(EULER)()()(2;0)(B201)(PRSC)
+(EULER)()()(4;0)(B202)(PRSC)
+(PDUM2,REI)()()(1122;-1)(B211)(U2)
+(PDUM2,REI)()()(3344;-1)(B222)(U2)
+(PDUM2,REI)()()(1324;-1)(B221)(U2)
+(PDUM2,REI)()()(1324;-1)(B212)(U2)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(M2)
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(M2)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(M2)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(M2)
 M2    3  #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
+()()(U1+U2)(102;0)(B3010)(M2)
+()()(U1+U2)(304;0)(B3020)(M2)
+()()(U1+U2)(102;0)(B31P1)(M2)
+()()(U1+U2)(304;0)(B32P2)(M2)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(M2)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(M2)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(M2)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(M2)
 M2    4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(M2)
+()()(U1+U2)(304;0)(B3020)(M2)
+()()(U1+U2)(102;0)(B31P1)(M2)
+()()(U1+U2)(304;0)(B32P2)(M2)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(M2)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(M2)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(M2)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(M2)
```

JACOBIANS

```
 RHO RHO  1 1  #  VARBL, VARDIF, SET, # OF TERMS, ALL DIRECTIONS
()()()(;1)(B200)()
 RHO RHO  2 1  #  VARBL, VARDIF, SET, # OF TERMS, ALL DIRECTIONS
+(-)()(U1+U2)(102;0)(B3010)()+(-)()(U1+U2)(304;0)(B3020)()
+()()(RHO)(1;0)(B3010)(UOR)+()()(RHO)(2;0)(B3010)(VOR)
+()()(RHO)(3;0)(B3020)(UOR)+()()(RHO)(4;0)(B3020)(VOR)
+(-,HBJ)()(RHO,HB1)(1122;-1)(B3110)(UMOR)
+(HBJ)()(RHO,HB1)(3344;-1)(B3220)(UMOR)
+(-,HBJ)()(RHO,HB1)(1324;-1)(B3120)(UMOR)
+(HBJ)()(RHO,HB1)(1324;-1)(B3210)(UMOR)
+(HBJ)()(UMAG,HB1)(1122;-1)(B3011)()+(HBJ)()(UMAG,HB1)(3344;-1)(B3022)()
+(HBJ)()(UMAG,HB1)(1324;-1)(B3012)()+(HBJ)()(UMAG,HB1)(1324;-1)(B3021)()
 RHO RHO  3 1  #  VARBL, VARDIF, SET, # OF TERMS, DIRECTION 1
()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(-)()(RHO)(1;0)(B3010)(UOR)+(-)()(RHO)(2;0)(B3010)(VOR)
+(-)()(RHO)(1;0)(B31P1)(UOR)+(-)()(RHO)(2;0)(B31P1)(VOR)
+(-)()(RHO)(3;0)(B3020)(UOR)+(-)()(RHO)(4;0)(B3020)(VOR)
+(-)()(RHO)(3;0)(B32P2)(UOR)+(-)()(RHO)(4;0)(B32P2)(VOR)
+(-,PHJI)()(RHO,HB1)(1122;-1)(B3110)(UMOR)
+(-,PHJI)()(RHO,HB1)(3344;-1)(B3220)(UMOR)
+(-,PHJI)()(RHO,HB1)(1324;-1)(B3120)(UMOR)
+(-,PHJI)()(RHO,HB1)(1324;-1)(B3210)(UMOR)
+(PHJI)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJI)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3021)()
 RHO RHO  4 1  #  VARBL, VARDIF, SET, # OF TERMS, DIRECTION 1
()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(-)()(RHO)(1;0)(B3010)(UOR)+(-)()(RHO)(2;0)(B3010)(VOR)
+(-)()(RHO)(1;0)(B31P1)(UOR)+(-)()(RHO)(2;0)(B31P1)(VOR)
+(-)()(RHO)(3;0)(B3020)(UOR)+(-)()(RHO)(4;0)(B3020)(VOR)
+(-)()(RHO)(3;0)(B32P2)(UOR)+(-)()(RHO)(4;0)(B32P2)(VOR)
+(PHJO)()(RHO,HB1)(1122;-1)(B3110)(UMOR)
+(PHJO)()(RHO,HB1)(3344;-1)(B3220)(UMOR)
+(PHJO)()(RHO,HB1)(1324;-1)(B3120)(UMOR)
+(PHJO)()(RHO,HB1)(1324;-1)(B3210)(UMOR)
+(PHJO)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJO)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3021)()

 ETOT ETOT  1 1  T
()()()(;1)(B200)()
 ETOT ETOT  2 1  T
+(-)()(U1+U2)(102;0)(B3010)()+(-)()(U1+U2)(304;0)(B3020)()
+(-GM1)()(U1+U2)(102;0)(B3010)()
+(-GM1)()(U1+U2)(304;0)(B3020)()
+(ZPEC,PEI)()()(1122;-1)(B211)()+(ZPEC,PEI)()()(3344;-1)(B222)()
+(ZPEC,PEI)()()(1324;-1)(B221)()+(ZPEC,PEI)()()(1324;-1)(B212)()
+(HBJ)()(UMAG,HB1)(1122;-1)(B3011)()+(HBJ)()(UMAG,HB1)(3344;-1)(B3022)()
+(HBJ)()(UMAG,HB1)(1324;-1)(B3012)()+(HBJ)()(UMAG,HB1)(1324;-1)(B3021)()
 ETOT ETOT  3  1  T
```

```
 ()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+(GM1)()(U1+U2)(102;0)(B3010)()+(GM1)()(U1+U2)(102;0)(B31P1)()
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(GM1)()(U1+U2)(304;0)(B3020)()+(GM1)()(U1+U2)(304;0)(B32P2)()
+(PHJI)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJI)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3021)()
 ETOT ETOT 4 1 T
 ()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+(GM1)()(U1+U2)(102;0)(B3010)()+(GM1)()(U1+U2)(102;0)(B31P1)()
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(GM1)()(U1+U2)(304;0)(B3020)()+(GM1)()(U1+U2)(304;0)(B32P2)()
+(PHJO)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJO)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3021)()

 M1 M1 1 1 T
()()()(;1)(B200)()
 M1 M1 2 1 T
+(-)()(M1)(1;0)(B3010)(OSRH)+(-)()(M1)(3;0)(B3020)(OSRH)
+(-,GM1 )()()(1;0)(B201)(U1)+(-,GM1 )()()(3;0)(B202)(U1)
+(-)()(U1+U2)(102;0)(B3010)()+(-)()(U1+U2)(304;0)(B3020)()
+(PDUM2,REI)()(OSRH)(1122;-1)(B3011)()
+(PDUM2,REI)()(OSRH)(3344;-1)(B3022)()
+(PDUM2,REI)()(OSRH)(1324;-1)(B3021)()
+(PDUM2,REI)()(OSRH)(1324;-1)(B3012)()
+(HBJ)()(M1,HB1)(1122;-1)(B3110)(UORU)
+(HBJ)()(M1,HB1)(3344;-1)(B3220)(UORU)
+(HBJ)()(M1,HB1)(1324;-1)(B3120)(UORU)
+(HBJ)()(M1,HB1)(1324;-1)(B3210)(UORU)
+(HBJ)()(UMAG,HB1)(1122;-1)(B3011)()+(HBJ)()(UMAG,HB1)(3344;-1)(B3022)()
+(HBJ)()(UMAG,HB1)(1324;-1)(B3012)()+(HBJ)()(UMAG,HB1)(1324;-1)(B3021)()
 M1 M1 3 1 T
 ()()(M1)(1;0)(B3010)(OSRH)+()()(M1)(1;0)(B31P1)(OSRH)
+()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+()()(M1)(3;0)(B3020)(OSRH)+()()(M1)(3;0)(B32P2)(OSRH)
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(PHJI)()(M1,HB1)(1122;-1)(B3110)(UORU)
+(PHJI)()(M1,HB1)(3344;-1)(B3220)(UORU)
+(PHJI)()(M1,HB1)(1324;-1)(B3120)(UORU)
+(PHJI)()(M1,HB1)(1324;-1)(B3210)(UORU)
+(PHJI)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJI)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3021)()
 M1 M1 4 1 T
 (FX2,GM1 )()()(1;0)(B201)(U1)+(FX2,GM1 )()()(3;0)(B202)(U1)
+()()(M1)(1;0)(B3010)(OSRH)+()()(M1)(1;0)(B31P1)(OSRH)
+()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+()()(M1)(3;0)(B3020)(OSRH)+()()(M1)(3;0)(B32P2)(OSRH)
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(PHJO)()(M1,HB1)(1122;-1)(B3110)(UORU)
+(PHJO)()(M1,HB1)(3344;-1)(B3220)(UORU)
+(PHJO)()(M1,HB1)(1324;-1)(B3120)(UORU)
```

```
+(PHJO)()(M1,HB1)(1324;-1)(B3210)(UORU)
+(PHJO)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJO)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3021)()
 M1 M1 8  1  T
(BTURB,REI)()(YPLS)(;1)(A200)()

 M2 M2 1 1  T
()()()(;1)(B200)()
 M2 M2 2 1  T
+(-,GM1 )()()(2;0)(B201)(U2)+(-,GM1 )()()(4;0)(B202)(U2)
+(-)()(M2)(2;0)(B3010)(OSRH)+(-)()(M2)(4;0)(B3020)(OSRH)
+(-)()(U1+U2)(102;0)(B3010)()+(-)()(U1+U2)(304;0)(B3020)()
+(PDUM2,REI)()(OSRH)(1122;-1)(B3011)()
+(PDUM2,REI)()(OSRH)(3344;-1)(B3022)()
+(PDUM2,REI)()(OSRH)(1324;-1)(B3021)()
+(PDUM2,REI)()(OSRH)(1324;-1)(B3012)()



+(HBJ)()(M2,HB1)(1122;-1)(B3110)(VORU)
+(HBJ)()(M2,HB1)(3344;-1)(B3220)(VORU)
+(HBJ)()(M2,HB1)(1324;-1)(B3120)(VORU)
+(HBJ)()(M2,HB1)(1324;-1)(B3210)(VORU)
+(HBJ)()(UMAG,HB1)(1122;-1)(B3011)()+(HBJ)()(UMAG,HB1)(3344;-1)(B3022)()
+(HBJ)()(UMAG,HB1)(1324;-1)(B3012)()+(HBJ)()(UMAG,HB1)(1324;-1)(B3021)()
 M2 M2 3  1  T
()()(M2)(2;0)(B3010)(OSRH)+()()(M2)(2;0)(B31P1)(OSRH)
+()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+()()(M2)(4;0)(B3020)(OSRH)+()()(M2)(4;0)(B32P2)(OSRH)
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(PHJI)()(M2,HB1)(1122;-1)(B3110)(VORU)
+(PHJI)()(M2,HB1)(3344;-1)(B3220)(VORU)
+(PHJI)()(M2,HB1)(1324;-1)(B3120)(VORU)
+(PHJI)()(M2,HB1)(1324;-1)(B3210)(VORU)
+(PHJI)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJI)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJI)()(UMAG,HB1)(1324;-1)(B3021)()
 M2 M2 4  1  T
(FX2,GM1 )()()(2;0)(B201)(U2)+(FX2,GM1 )()()(4;0)(B202)(U2)
+()()(M2)(2;0)(B3010)(OSRH)+()()(M2)(2;0)(B31P1)(OSRH)
+()()(U1+U2)(102;0)(B3010)()+()()(U1+U2)(102;0)(B31P1)()
+()()(M2)(4;0)(B3020)(OSRH)+()()(M2)(4;0)(B32P2)(OSRH)
+()()(U1+U2)(304;0)(B3020)()+()()(U1+U2)(304;0)(B32P2)()
+(PHJO)()(M2,HB1)(1122;-1)(B3110)(VORU)
+(PHJO)()(M2,HB1)(3344;-1)(B3220)(VORU)
+(PHJO)()(M2,HB1)(1324;-1)(B3120)(VORU)
+(PHJO)()(M2,HB1)(1324;-1)(B3210)(VORU)
+(PHJO)()(UMAG,HB1)(1122;-1)(B3011)()
+(PHJO)()(UMAG,HB1)(3344;-1)(B3022)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3012)()
+(PHJO)()(UMAG,HB1)(1324;-1)(B3021)()
 M2 M2 8  1  T
(BTURB,REI)()(YPLS)(;1)(A200)()
```

```
 RHO M1 2  1  T
(-)()(RHO)(1;0)(B3010)(OSRH)+(-)()(RHO)(3;0)(B3020)(OSRH)
+(HBJ)()(RHO,HB1)(1122;-1)(B3110)(UORU)
+(HBJ)()(RHO,HB1)(3344;-1)(B3220)(UORU)
+(HBJ)()(RHO,HB1)(1324;-1)(B3120)(UORU)
+(HBJ)()(RHO,HB1)(1324;-1)(B3210)(UORU)
 RHO M1 3  1  T
()()(RHO)(1;0)(B3010)(OSRH)+()()(RHO)(1;0)(B31P1)(OSRH)
+()()(RHO)(3;0)(B3020)(OSRH)+()()(RHO)(3;0)(B32P2)(OSRH)
+(PHJI)()(RHO,HB1)(1122;-1)(B3110)(UORU)
+(PHJI)()(RHO,HB1)(3344;-1)(B3220)(UORU)
+(PHJI)()(RHO,HB1)(1324;-1)(B3120)(UORU)
+(PHJI)()(RHO,HB1)(1324;-1)(B3210)(UORU)
 RHO M1 4  1  T
()()(RHO)(1;0)(B3010)(OSRH)+()()(RHO)(1;0)(B31P1)(OSRH)
+()()(RHO)(3;0)(B3020)(OSRH)+()()(RHO)(3;0)(B32P2)(OSRH)
+(PHJO)()(RHO,HB1)(1122;-1)(B3110)(UORU)
+(PHJO)()(RHO,HB1)(3344;-1)(B3220)(UORU)
+(PHJO)()(RHO,HB1)(1324;-1)(B3120)(UORU)
+(PHJO)()(RHO,HB1)(1324;-1)(B3210)(UORU)

 RHO M2 2  1  T
(-)()(RHO)(2;0)(B3010)(OSRH)+(-)()(RHO)(4;0)(B3020)(OSRH)
+(HBJ)()(RHO,HB1)(1122;-1)(B3110)(VORU)
+(HBJ)()(RHO,HB1)(3344;-1)(B3220)(VORU)
+(HBJ)()(RHO,HB1)(1324;-1)(B3120)(VORU)
+(HBJ)()(RHO,HB1)(1324;-1)(B3210)(VORU)
 RHO M2 3  1  T
()()(RHO)(2;0)(B3010)(OSRH)+()()(RHO)(2;0)(B31P1)(OSRH)
+()()(RHO)(4;0)(B3020)(OSRH)+()()(RHO)(4;0)(B32P2)(OSRH)
+(PHJI)()(RHO,HB1)(1122;-1)(B3110)(VORU)
+(PHJI)()(RHO,HB1)(3344;-1)(B3220)(VORU)
+(PHJI)()(RHO,HB1)(1324;-1)(B3120)(VORU)
+(PHJI)()(RHO,HB1)(1324;-1)(B3210)(VORU)
 RHO M2 4  1  T
()()(RHO)(2;0)(B3010)(OSRH)+()()(RHO)(2;0)(B31P1)(OSRH)
+()()(RHO)(4;0)(B3020)(OSRH)+()()(RHO)(4;0)(B32P2)(OSRH)
+(PHJO)()(RHO,HB1)(1122;-1)(B3110)(VORU)
+(PHJO)()(RHO,HB1)(3344;-1)(B3220)(VORU)
+(PHJO)()(RHO,HB1)(1324;-1)(B3120)(VORU)
+(PHJO)()(RHO,HB1)(1324;-1)(B3210)(VORU)

 ETOT RHO 2 1  T
()()(EP)(1;0)(B3010)(UOR)+()()(EP)(2;0)(B3010)(VOR)
+()()(EP)(3;0)(B3020)(UOR)+()()(EP)(4;0)(B3020)(VOR)
+(HGMMKI)()(U1+U2)(102;0)(B3010)(UVWS)
+(HGMMKI)()(U1+U2)(304;0)(B3020)(UVWS)
+(HBJ)()(ETOT,HB1)(1122;-1)(B3110)(UMOR)
+(HBJ)()(ETOT,HB1)(3344;-1)(B3220)(UMOR)
+(HBJ)()(ETOT,HB1)(1324;-1)(B3120)(UMOR)
+(HBJ)()(ETOT,HB1)(1324;-1)(B3210)(UMOR)
 ETOT RHO 3 1  T
(-)()(EP)(1;0)(B3010)(UOR)+(-)()(EP)(2;0)(B3010)(VOR)
+(-)()(EP)(3;0)(B3020)(UOR)+(-)()(EP)(4;0)(B3020)(VOR)
+(-)()(EP)(1;0)(B31P1)(UOR)+(-)()(EP)(2;0)(B31P1)(VOR)
```

```
+(-)()(EP)(3;0)(B32P2)(UOR)+(-)()(EP)(4;0)(B32P2)(VOR)
+(-,HGMMKI)()(U1+U2)(102;0)(B3010)(UVWS)
+(-,HGMMKI)()(U1+U2)(304;0)(B3020)(UVWS)
+(-,HGMMKI)()(U1+U2)(102;0)(B31P1)(UVWS)
+(-,HGMMKI)()(U1+U2)(304;0)(B32P2)(UVWS)
+(PHJI)()(ETOT,HB1)(1122;-1)(B3110)(UMOR)
+(PHJI)()(ETOT,HB1)(3344;-1)(B3220)(UMOR)
+(PHJI)()(ETOT,HB1)(1324;-1)(B3120)(UMOR)
+(PHJI)()(ETOT,HB1)(1324;-1)(B3210)(UMOR)
 ETOT RHO 4 1  T
(-)()(EP)(1;0)(B3010)(UOR)+(-)()(EP)(2;0)(B3010)(VOR)
+(-)()(EP)(3;0)(B3020)(UOR)+(-)()(EP)(4;0)(B3020)(VOR)
+(-)()(EP)(1;0)(B31P1)(UOR)+(-)()(EP)(2;0)(B31P1)(VOR)
+(-)()(EP)(3;0)(B32P2)(UOR)+(-)()(EP)(4;0)(B32P2)(VOR)
+(-,HGMMKI)()(U1+U2)(102;0)(B3010)(UVWS)
+(-,HGMMKI)()(U1+U2)(304;0)(B3020)(UVWS)
+(NF2,-,HGMMKI)()(U1+U2)(102;0)(B31P1)(UVWS)
+(NF2,-,HGMMKI)()(U1+U2)(304;0)(B32P2)(UVWS)
+(PHJO)()(ETOT,HB1)(1122;-1)(B3110)(UMOR)
+(PHJO)()(ETOT,HB1)(3344;-1)(B3220)(UMOR)
+(PHJO)()(ETOT,HB1)(1324;-1)(B3120)(UMOR)
+(PHJO)()(ETOT,HB1)(1324;-1)(B3210)(UMOR)

 ETOT M1 2  1  T
(-)()(EP)(1;0)(B3010)(OSRH)+(-)()(EP)(3;0)(B3020)(OSRH)
+(HBJ)()(ETOT,HB1)(1122;-1)(B3110)(UORU)
+(HBJ)()(ETOT,HB1)(3344;-1)(B3220)(UORU)
+(HBJ)()(ETOT,HB1)(1324;-1)(B3120)(UORU)
+(HBJ)()(ETOT,HB1)(1324;-1)(B3210)(UORU)
+(GMMKI)()(U1+U2)(102;0)(B3010)(U1)
+(GMMKI)()(U1+U2)(304;0)(B3020)(U1)
 ETOT M1 3  1  T
()()(EP)(1;0)(B3010)(OSRH)+()()(EP)(1;0)(B31P1)(OSRH)
+()()(EP)(3;0)(B3020)(OSRH)+()()(EP)(3;0)(B32P2)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(B3010)(U1)
+(-GMMKI)()(U1+U2)(304;0)(B3020)(U1)
+(-GMMKI)()(U1+U2)(102;0)(B31P1)(U1)
+(-GMMKI)()(U1+U2)(304;0)(B32P2)(U1)


+(PHJI)()(ETOT,HB1)(1122;-1)(B3110)(UORU)
+(PHJI)()(ETOT,HB1)(3344;-1)(B3220)(UORU)
+(PHJI)()(ETOT,HB1)(1324;-1)(B3120)(UORU)
+(PHJI)()(ETOT,HB1)(1324;-1)(B3210)(UORU)
 ETOT M1 4  1  T
()()(EP)(1;0)(B3010)(OSRH)+()()(EP)(1;0)(B31P1)(OSRH)
+()()(EP)(3;0)(B3020)(OSRH)+()()(EP)(3;0)(B32P2)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(B3010)(U1)
+(-GMMKI)()(U1+U2)(304;0)(B3020)(U1)
+(NF2,-GMMKI)()(U1+U2)(102;0)(B31P1)(U1)
+(NF2,-GMMKI)()(U1+U2)(304;0)(B32P2)(U1)
+(PHJO)()(ETOT,HB1)(1122;-1)(B3110)(UORU)
+(PHJO)()(ETOT,HB1)(3344;-1)(B3220)(UORU)
+(PHJO)()(ETOT,HB1)(1324;-1)(B3120)(UORU)
+(PHJO)()(ETOT,HB1)(1324;-1)(B3210)(UORU)
```

```
 ETOT M2 2  1  T
(-)()(EP)(2;0)(B3010)(OSRH)+(-)()(EP)(4;0)(B3020)(OSRH)
+(HBJ)()(ETOT,HB1)(1122;-1)(B3110)(VORU)
+(HBJ)()(ETOT,HB1)(3344;-1)(B3220)(VORU)
+(HBJ)()(ETOT,HB1)(1324;-1)(B3120)(VORU)
+(HBJ)()(ETOT,HB1)(1324;-1)(B3210)(VORU)
+(GMMKI)()(U1+U2)(102;0)(B3010)(U2)
+(GMMKI)()(U1+U2)(304;0)(B3020)(U2)
 ETOT M2 3  1  T
()()(EP)(2;0)(B3010)(OSRH)+()()(EP)(2;0)(B31P1)(OSRH)
+()()(EP)(4;0)(B3020)(OSRH)+()()(EP)(4;0)(B32P2)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(B3010)(U2)
+(-GMMKI)()(U1+U2)(304;0)(B3020)(U2)
+(-GMMKI)()(U1+U2)(102;0)(B31P1)(U2)
+(-GMMKI)()(U1+U2)(304;0)(B32P2)(U2)
+(PHJI)()(ETOT,HB1)(1122;-1)(B3110)(VORU)
+(PHJI)()(ETOT,HB1)(3344;-1)(B3220)(VORU)
+(PHJI)()(ETOT,HB1)(1324;-1)(B3120)(VORU)
+(PHJI)()(ETOT,HB1)(1324;-1)(B3210)(VORU)
 ETOT M2 4  1  T
()()(EP)(2;0)(B3010)(OSRH)+()()(EP)(2;0)(B31P1)(OSRH)
+()()(EP)(4;0)(B3020)(OSRH)+()()(EP)(4;0)(B32P2)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(B3010)(U2)
+(-GMMKI)()(U1+U2)(304;0)(B3020)(U2)
+(NF2,-GMMKI)()(U1+U2)(102;0)(B31P1)(U2)
+(NF2,-GMMKI)()(U1+U2)(304;0)(B32P2)(U2)
+(PHJO)()(ETOT,HB1)(1122;-1)(B3110)(VORU)
+(PHJO)()(ETOT,HB1)(3344;-1)(B3220)(VORU)
+(PHJO)()(ETOT,HB1)(1324;-1)(B3120)(VORU)
+(PHJO)()(ETOT,HB1)(1324;-1)(B3210)(VORU)

 M1 RHO 2 1  T
(GMH)()()(1;0)(B201)(UVWS)+(GMH)()()(3;0)(B202)(UVWS)
+()()(M1)(1;0)(B3010)(UOR)+()()(M1)(2;0)(B3010)(VOR)
+()()(M1)(3;0)(B3020)(UOR)+()()(M1)(4;0)(B3020)(VOR)
+(HBJ)()(M1,HB1)(1122;-1)(B3110)(UMOR)
+(HBJ)()(M1,HB1)(3344;-1)(B3220)(UMOR)
+(HBJ)()(M1,HB1)(1324;-1)(B3120)(UMOR)
+(HBJ)()(M1,HB1)(1324;-1)(B3210)(UMOR)
 M1 RHO 3  1  T
(-)()(M1)(1;0)(B3010)(UOR)+(-)()(M1)(2;0)(B3010)(VOR)
+(-)()(M1)(1;0)(B31P1)(UOR)+(-)()(M1)(2;0)(B31P1)(VOR)
+(-)()(M1)(3;0)(B3020)(UOR)+(-)()(M1)(4;0)(B3020)(VOR)
+(-)()(M1)(3;0)(B32P2)(UOR)+(-)()(M1)(4;0)(B32P2)(VOR)
+(PHJI)()(M1,HB1)(1122;-1)(B3110)(UMOR)
+(PHJI)()(M1,HB1)(3344;-1)(B3220)(UMOR)
+(PHJI)()(M1,HB1)(1324;-1)(B3120)(UMOR)
+(PHJI)()(M1,HB1)(1324;-1)(B3210)(UMOR)
 M1 RHO 4 1  T
(FX2,-GMH)()()(1;0)(B201)(UVWS)+(FX2,-GMH)()()(3;0)(B202)(UVWS)
+(-)()(M1)(1;0)(B3010)(UOR)+(-)()(M1)(2;0)(B3010)(VOR)
+(-)()(M1)(1;0)(B31P1)(UOR)+(-)()(M1)(2;0)(B31P1)(VOR)
+(-)()(M1)(3;0)(B3020)(UOR)+(-)()(M1)(4;0)(B3020)(VOR)
+(-)()(M1)(3;0)(B32P2)(UOR)+(-)()(M1)(4;0)(B32P2)(VOR)
+(PHJO)()(M1,HB1)(1122;-1)(B3110)(UMOR)
+(PHJO)()(M1,HB1)(3344;-1)(B3220)(UMOR)
```

```
+(PHJO)()(M1,HB1)(1324;-1)(B3120)(UMOR)
+(PHJO)()(M1,HB1)(1324;-1)(B3210)(UMOR)

 M1 ETOT 2  1  T
(GM1MK)()()(1;0)(B201)()+(GM1MK)()()(3;0)(B202)()
 M1 ETOT 4  1  T
(FX2,-GM1MK)()()(1;0)(B201)()+(FX2,-GM1MK)()()(3;0)(B202)()

 M1 M2 2  1  T
(-,GM1   )()()(1;0)(B201)(U2)+(-,GM1   )()()(3;0)(B202)(U2)
+(-)()(M1)(2;0)(B3010)(OSRH)+(-)()(M1)(4;0)(B3020)(OSRH)
+(HBJ)()(M1,HB1)(1122;-1)(B3110)(VORU)
+(HBJ)()(M1,HB1)(3344;-1)(B3220)(VORU)
+(HBJ)()(M1,HB1)(1324;-1)(B3120)(VORU)
+(HBJ)()(M1,HB1)(1324;-1)(B3210)(VORU)
 M1 M2  3  1  T
()()(M1)(2;0)(B3010)(OSRH)+()()(M1)(2;0)(B31P1)(OSRH)
+()()(M1)(4;0)(B3020)(OSRH)+()()(M1)(4;0)(B32P2)(OSRH)
+(PHJI)()(M1,HB1)(1122;-1)(B3110)(VORU)
+(PHJI)()(M1,HB1)(3344;-1)(B3220)(VORU)
+(PHJI)()(M1,HB1)(1324;-1)(B3120)(VORU)
+(PHJI)()(M1,HB1)(1324;-1)(B3210)(VORU)
 M1 M2  4  1  T
(FX2,GM1   )()()(1;0)(B201)(U2)+(FX2,GM1   )()()(3;0)(B202)(U2)
+()()(M1)(2;0)(B3010)(OSRH)+()()(M1)(2;0)(B31P1)(OSRH)
+()()(M1)(4;0)(B3020)(OSRH)+()()(M1)(4;0)(B32P2)(OSRH)
+(PHJO)()(M1,HB1)(1122;-1)(B3110)(VORU)
+(PHJO)()(M1,HB1)(3344;-1)(B3220)(VORU)
+(PHJO)()(M1,HB1)(1324;-1)(B3120)(VORU)
+(PHJO)()(M1,HB1)(1324;-1)(B3210)(VORU)

 M2 RHO 2  1  T
(GMH)()()(2;0)(B201)(UVWS)+(GMH)()()(4;0)(B202)(UVWS)
+()()(M2)(1;0)(B3010)(UOR)+()()(M2)(2;0)(B3010)(VOR)
+()()(M2)(3;0)(B3020)(UOR)+()()(M2)(4;0)(B3020)(VOR)
+(HBJ)()(M2,HB1)(1122;-1)(B3110)(UMOR)
+(HBJ)()(M2,HB1)(3344;-1)(B3220)(UMOR)
+(HBJ)()(M2,HB1)(1324;-1)(B3120)(UMOR)
+(HBJ)()(M2,HB1)(1324;-1)(B3210)(UMOR)
 M2 RHO 3  1  T
(-)()(M2)(1;0)(B3010)(UOR)+(-)()(M2)(2;0)(B3010)(VOR)
+(-)()(M2)(1;0)(B31P1)(UOR)+(-)()(M2)(2;0)(B31P1)(VOR)
+(-)()(M2)(3;0)(B3020)(UOR)+(-)()(M2)(4;0)(B3020)(VOR)
+(-)()(M2)(3;0)(B32P2)(UOR)+(-)()(M2)(4;0)(B32P2)(VOR)
+(PHJI)()(M2,HB1)(1122;-1)(B3110)(UMOR)
+(PHJI)()(M2,HB1)(3344;-1)(B3220)(UMOR)
+(PHJI)()(M2,HB1)(1324;-1)(B3120)(UMOR)
+(PHJI)()(M2,HB1)(1324;-1)(B3210)(UMOR)
 M2 RHO 4 1  T
(FX2,-GMH)()()(2;0)(B201)(UVWS)+(FX2,-GMH)()()(4;0)(B202)(UVWS)
+(-)()(M2)(1;0)(B3010)(UOR)+(-)()(M2)(2;0)(B3010)(VOR)
+(-)()(M2)(1;0)(B31P1)(UOR)+(-)()(M2)(2;0)(B31P1)(VOR)
+(-)()(M2)(3;0)(B3020)(UOR)+(-)()(M2)(4;0)(B3020)(VOR)
+(-)()(M2)(3;0)(B32P2)(UOR)+(-)()(M2)(4;0)(B32P2)(VOR)
+(PHJO)()(M2,HB1)(1122;-1)(B3110)(UMOR)
+(PHJO)()(M2,HB1)(3344;-1)(B3220)(UMOR)
```

```
 +(PHJO)()(M2,HB1)(1324;-1)(B3120)(UMOR)
 +(PHJO)()(M2,HB1)(1324;-1)(B3210)(UMOR)

  M2 ETOT 2  1  T
 (GM1MK)()()(2;0)(B201)()+(GM1MK)()()(4;0)(B202)()
  M2 ETOT 4  1  T
 (FX2,-GM1MK)()()(2;0)(B201)()
+(FX2,-GM1MK)()()(4;0)(B202)()

  M2 M1 2  1  T
 (-,GM1   )()()(2;0)(B201)(U1)
+(-,GM1   )()()(4;0)(B202)(U1)
+(-)()(M2)(1;0)(B3010)(OSRH)
+(-)()(M2)(3;0)(B3020)(OSRH)
+(HBJ)()(M2,HB1)(1122;-1)(B3110)(UORU)
+(HBJ)()(M2,HB1)(3344;-1)(B3220)(UORU)
+(HBJ)()(M2,HB1)(1324;-1)(B3120)(UORU)
+(HBJ)()(M2,HB1)(1324;-1)(B3210)(UORU)
  M2 M1 3  1  T
 ()()(M2)(1;0)(B3010)(OSRH)+()()(M2)(1;0)(B31P1)(OSRH)
+()()(M2)(3;0)(B3020)(OSRH)+()()(M2)(3;0)(B32P2)(OSRH)
+(PHJI)()(M2,HB1)(1122;-1)(B3110)(UORU)
+(PHJI)()(M2,HB1)(3344;-1)(B3220)(UORU)
+(PHJI)()(M2,HB1)(1324;-1)(B3120)(UORU)
+(PHJI)()(M2,HB1)(1324;-1)(B3210)(UORU)
  M2 M1 4  1  T
 (FX2,GM1   )()()(2;0)(B201)(U1)
+(FX2,GM1   )()()(4;0)(B202)(U1)
+()()(M2)(1;0)(B3010)(OSRH)+()()(M2)(1;0)(B31P1)(OSRH)
+()()(M2)(3;0)(B3020)(OSRH)+()()(M2)(3;0)(B32P2)(OSRH)
+(PHJO)()(M2,HB1)(1122;-1)(B3110)(UORU)
+(PHJO)()(M2,HB1)(3344;-1)(B3220)(UORU)
+(PHJO)()(M2,HB1)(1324;-1)(B3120)(UORU)
+(PHJO)()(M2,HB1)(1324;-1)(B3210)(UORU)

GROUP FREQUENCY
 1
SOLUTION TYPE
 DELTA_Q
 LU_FACTORIZATION_INCOMPLETE
 IMPLICIT_EULER

END
```

APPENDIX B

TWS^h FE *REMI* algorithm, $d = 1,2,3$

The generic element level TWS^h matrix statement is:

$$\{FQ\}_e^p = [M]_e\left\{Q_{n+1}^p - Qn\right\}_e + \Delta t\left(\theta\{RQ\}_{n+1}^p + \{RQ\}_n\right)_e \tag{5.43}$$

with contributions

$$[M]_e\left\{Q_{n+1}^p - Qn\right\}_e = \int_{\Omega e}\{N\}\{N\}^T d\tau\left\{Q_{n+1}^p - Qn\right\}_e \tag{5.4}$$

$$\begin{aligned}\{RQ\}_e = &-\int_{\Omega_e}\frac{\partial\{N\}}{\partial x_j}\left(f_j - f_j^v - \beta h u_j \hat{u}_k \frac{\partial q}{\partial x_k}\right)_e d\tau \\ &+\oint_{\partial\Omega_e}\{N\}\left(f_j - f_j^v - \beta h u_j \hat{u}_k \frac{\partial q}{\partial x_k}\right)_e \hat{n}_j \, d\sigma\end{aligned} \tag{5.5}$$

For α=0=γ in the TWS^h procedure, (5.4) is universal for all $\{Q\}$ in the form

$$[M]_e\left\{Q_{n+1}^p - Q_n\right\}_e = [\mathrm{M200E}]\left\{Q_{n+1}^p - Q_n\right\}_e \tag{5.44}$$

All surface integrals created in the residual (5.5) are handled as discussed, hence the residual expressions contain only terms on the generic FE domain Ω_e

$$\{\mathrm{RR}\}_e = -[M2J0]_e\{\mathrm{M}J\}_e + \beta R\hat{\mathrm{U}}Ke\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e\{\mathrm{RHO}\}_e \tag{5.45}$$

$$\begin{aligned}\{\mathrm{RE}\}_e = &-\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]_e\{\mathrm{ETOT} + (\mathrm{Eu}/\mathrm{Ec})\mathrm{PRES}\}_e \\ &+ \quad \mathrm{Pe}^{-1}[\mathrm{M2}KK]_e\{\mathrm{TEMP}\} + \beta_e\hat{\mathrm{U}}K\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e\{\mathrm{ETOT}\}_e\end{aligned} \tag{5.46}$$

$$\begin{aligned}\{\mathrm{RM1}\}_e = &-\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]_e\{\mathrm{M1}\}_e + \mathrm{Eu}[\mathrm{M201}]_e\{\mathrm{PRES}\}_e \\ &+\mathrm{Re}^{-1}[\mathrm{M2}KK]_e\{\mathrm{U1}\}_e + \mathrm{Re}^{-1}[\mathrm{M21}K]_e\{\mathrm{U}K/3\}_e \\ &+\beta_1\hat{\mathrm{U}}K_e\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e\{M1\}_e\end{aligned} \tag{5.47}$$

$$\begin{aligned}\{\mathrm{RM2}\}_e = &-\{\mathrm{U}J\}_e^T[M30J0]_e\{\mathrm{M2}\}_e + \mathrm{Eu}[\mathrm{M202}]_e\{\mathrm{PRES}\}_e \\ &+\mathrm{Re}^{-1}[\mathrm{M2}KK]_e\{\mathrm{U2}\}_e + \mathrm{Re}^{-1}[\mathrm{M22}K]_e\{\mathrm{U}K/3\}_e \\ &+\beta_2\hat{\mathrm{U}}K_e\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e\{M2\}_e\end{aligned} \tag{5.48}$$

$$\begin{aligned}\{\mathrm{RM3}\}_e &= -\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]_e\{\mathrm{M3}\}_e + \mathrm{Eu}[\mathrm{M203}]_e\{\mathrm{PRES}\}_e \\ &+\mathrm{Re}^{-1}[\mathrm{M2}KK]_e\{\mathrm{U3}\}_e + \mathrm{Re}^{-1}[\mathrm{M23}K]_e\{\mathrm{U}K/3\}_e \\ &+\beta 3\hat{\mathrm{U}}Ke\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e\{M3\}_e\end{aligned} \tag{5.49}$$

The Newton jacobian (5.11) is constructed as the assembly of:

$$[JAC]_e \equiv \frac{\partial\{FQ\}_e}{\partial\{Q\}_e} = [M]_e + \theta\Delta t\frac{\partial\{RQ\}_e}{\partial\{Q\}_e} \tag{5.12}$$

The $[M]_e$ term is common to all residuals. All other derivatives of $\{Q\}_e$ require use of the chain rule.

The density jacobians are:

$$\begin{aligned}\frac{\partial\{\mathrm{RR}\}_e}{\partial\{\mathrm{RHO}\}_e} &\equiv [\mathrm{RR,R}]_e = \beta R\hat{\mathrm{U}}Ke\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e \\ &-2\beta R\hat{\mathrm{U}}Ke\{\mathrm{RHO}\}_e^T[\mathrm{M3}JK0]_e\lceil \mathrm{U}J/\mathrm{RHO}\rfloor_e\end{aligned} \tag{5.52}$$

$$[\mathrm{RR,M}I]_e = [\mathrm{M2}I0]_e + 2\beta R\hat{\mathrm{U}}Ke\{\mathrm{RHO}\}_e^T[\mathrm{M3}KJ0]_e\lceil \mathrm{OSRH}\rfloor_e \tag{5.55}$$

Since "I" is a free tensor index in (5.55), there are d expressions.

The energy jacobians are:

$$\begin{aligned}[\mathrm{RE,E}]_e &= -\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]_e - (\gamma-1)\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]_e \\ &+\frac{(\gamma-1)\mathrm{Ec}}{\mathrm{PeEu}}[\mathrm{M2}KK]_e\lceil \mathrm{OSRH}\rfloor_e + \beta E\hat{\mathrm{U}}Ke\{\mathrm{U}J\}_e^T[\mathrm{M30}KJ]_e\end{aligned} \tag{5.59}$$

$$\begin{aligned}[\mathrm{RE,R}]_e &= \{\mathrm{ETOT} + (\mathrm{Ec}/\mathrm{Eu})\mathrm{PRES}\}_e^T[\mathrm{M30}J0]_e\lceil \mathrm{U}J\mathrm{SR}\rfloor_e \\ &+\frac{(\gamma-1)}{2\mathrm{Ec}}\{\mathrm{U}J\}_e^T[\mathrm{M30}J0]_e\lceil \mathrm{UKUK}\rfloor_e - \frac{1}{\mathrm{Pe}}[\mathrm{M2}KK]_e\lceil \mathrm{TSRH}\rfloor_e \\ &-2\beta E\hat{\mathrm{U}}Ke\{\mathrm{ETOT}\}_e^T[\mathrm{M3}JK0]_e\lceil \mathrm{U}J\mathrm{SR}\rfloor_e\end{aligned} \tag{5.64}$$

$$[\mathrm{RE},\mathrm{M}I]_e = \frac{(\gamma-1)}{\mathrm{Ec}}\{\mathrm{U}J\}_e^T[\mathrm{M}30J0]_e\lceil \mathrm{U}I\rceil_e$$
$$-\{\mathrm{ETOT}+(\mathrm{Eu}/\mathrm{Ec})\mathrm{PRES}\}_e^T[\mathrm{M}30I0]_e\lceil \mathrm{OSRH}\rfloor_e$$
$$\frac{-(\gamma-1)}{\mathrm{PeEu}}[\mathrm{M}2KK]_e\lceil \mathrm{U}I\mathrm{SR}\rfloor_e + 2\hat{\mathrm{U}}Ke\{\mathrm{ETOT}\}_e^T[\mathrm{M}3KI0]_e\lceil \mathrm{OSRH}\rfloor_e \tag{5.68}$$

The momentum jacobians are:

$$[\mathrm{RM}I,\mathrm{R}]_e = \{\mathrm{M}I\}_e^T[\mathrm{M}30J0]_e\lceil \mathrm{U}J\mathrm{SR}\rfloor_e + \frac{(\gamma-1)}{2}[\mathrm{M}20I]_e\lceil \mathrm{U}K\mathrm{U}K\rfloor_e$$
$$-\frac{1}{\mathrm{Re}}[\mathrm{M}2KK]_e\lceil \mathrm{U}I\mathrm{SR}\rfloor_e - \frac{1}{3\mathrm{Re}}[\mathrm{M}2IK]_e\lceil \mathrm{U}K\mathrm{SR}\rfloor_e$$
$$-2\beta\underline{I}\hat{\mathrm{U}}Ke\{\mathrm{M}I\}_e^T[\mathrm{M}3JK0]_e\lceil \mathrm{U}J\mathrm{SR}\rfloor_e \tag{5.70}$$

$$[\mathrm{RM}I,\mathrm{E}]_e = \mathrm{Eu}[\mathrm{M}20I]_e\frac{\partial\{\mathrm{PRES}\}_e}{\partial\{\mathrm{ETOT}\}_e} = (\gamma-1)\mathrm{Ec}[\mathrm{M}20I]_e \tag{5.71}$$

The self-coupling jacobian contributions are

$$[\mathrm{RM}\underline{I},\mathrm{M}\underline{I}]_e = -\{\mathrm{U}J\}_e^T[\mathrm{M}30J0]_e - \{\mathrm{M}\underline{I}\}_e^T\{\mathrm{M}30\underline{I}0\}_e\lceil \mathrm{OSRH}\rfloor_e$$
$$-(\gamma-1)[\mathrm{M}20\underline{I}]_e\lceil \mathrm{U}\underline{I}\rfloor_e + \frac{1}{\mathrm{Re}}[\mathrm{M}2KK]_e\lceil \mathrm{OSRH}\rfloor_e$$
$$+\frac{1}{3\mathrm{Re}}[\mathrm{M}2\underline{I}\underline{I}]_e\lceil \mathrm{OSRH}\rfloor_e + \beta\underline{I}\hat{\mathrm{U}}Ke\{\mathrm{U}J\}_e^T[\mathrm{M}30KJ]_e$$
$$+\beta\underline{I}\hat{\mathrm{U}}Ke\{\mathrm{M}\underline{I}\}_e^T[\mathrm{M}3IK0]_e\lceil \mathrm{OSRH}\rfloor_e \tag{5.73}$$

The non-self coupled jacobians are:

$$[\mathrm{RM}I,\mathrm{M}J]_e = -\{\mathrm{M}I\}_e^T[\mathrm{M}30J0]_e\lceil \mathrm{OSRH}\rfloor_e$$
$$-(\gamma-1)[\mathrm{M}20I]_e\lceil \mathrm{U}J\rfloor_e + \frac{1}{3\mathrm{Re}}[\mathrm{M}2IJ]_e\lceil \mathrm{OSRH}\rfloor_e$$
$$+2\beta\underline{I}\hat{\mathrm{U}}Ke\{\mathrm{M}I\}_e^T[\mathrm{M}3JK0]_e\lceil \mathrm{OSRH}\rfloor_e \tag{5.74}$$

APPENDIX C
AKCESS.AERO template, *d*=2 quasi-Newton jacobian

INTEGRATION FACTORS

```
 INITIAL_TIME
 FINAL_TIME
 PROBLEM_CONVERGENCE_CRITERIA
 MAXIMUM_CHANGE_IN_Q__(DQ)
 INITIAL_TIME_STEP
 TIME_STEP_MULTIPLIER
 MAXIMUM_TIME_STEP
 CRITERIA_TO_RAISE_MAX_TIME_STEP
 MAXIMUM_NUMBER_OF_STEPS
 MAXIMUM_NUMBER_OF_ITERATIONS_PER_STEP
 ITERATION_CONVERGENCE_CRITERIA
 THETA__IMPLICITNESS_FACTOR
 MAXIMUM_VALUE_OF_ANY_DELTA_Q

TRANSFORMATION ARRAYS
   ETKJ    1.
   DETJ    1.
#    DETE    0.

BOUNDARY CONDITIONS
       RHO  ETOT  M1  M2  PRSC    # ORDER
 DIR_RT    D    D   0   0    0    # DIRICHLET
 DIR_M1    0    0   D   0    0    # DIRICHLET
 DIR_M2    0    0   0   D    0    # DIRICHLET
 NO_SLIP   0    0   D   D    0    # NO SLIP WALL
 DIR_PRS   0    0   0   0    D    # DIRICHLET
 THR_IN  [3 -3] [3 -3] [3 -3] [3 -3]  0   # THROUGH FLOW
 THR_OUT [4 -4] [4 -4] [4 -4] [4 -4]  0   # THROUGH FLOW
 BLANK     D    D   D   D    D   # NO SLIP WALL

TITLE          **** TEMPLATE FILE TEMP.CNS2D.REMI ****
 CNS2D  TWS ALGORITHM,  TENSOR MATRIX A JACOBIANS (12/23/93)

RESIDUALS

 RHO   1  #  VARIABLE, SET NO., --- [M] * {[RHO.NEW] - [RHO.OLD]}
()()()(;1)(B200)(-RHO)
 RHO   2  #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
(-)()(U1+U2)(102;0)(B3010)(RHO)
+(-)()(U1+U2)(304;0)(B3020)(RHO)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(RHO)
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(RHO)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(RHO)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(RHO)
 RHO   3  #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
()()(U1+U2)(102;0)(B3010)(RHO)
+()()(U1+U2)(304;0)(B3020)(RHO)
+()()(U1+U2)(102;0)(B31P1)(RHO)
```


PRECEDING PAGE BLANK NOT FILMED

```
+()()(U1+U2)(304;0)(B32P2)(RHO)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(RHO)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(RHO)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(RHO)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(RHO)
 RHO   4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(RHO)
+()()(U1+U2)(304;0)(B3020)(RHO)
+()()(U1+U2)(102;0)(B31P1)(RHO)
+()()(U1+U2)(304;0)(B32P2)(RHO)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(RHO)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(RHO)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(RHO)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(RHO)

 ETOT   1  #  VARIABLE, SET NO., --- [M] * {[ETOT.NEW] - [ETOT.OLD]}
 ()()()(;1)(B200)(-ETOT)
 ETOT   2  #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
 (-)()(U1+U2)(102;0)(B3010)(EP)
+(-)()(U1+U2)(304;0)(B3020)(EP)
+(PDUM2,PEI)()()(1122;-1)(B211)(TEMP)
+(PDUM2,PEI)()()(3344;-1)(B222)(TEMP)
+(PDUM2,PEI)()()(1324;-1)(B221)(TEMP)
+(PDUM2,PEI)()()(1324;-1)(B212)(TEMP)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(ETOT)
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(ETOT)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(ETOT)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(ETOT)
 ETOT   3  #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(EP)
+()()(U1+U2)(304;0)(B3020)(EP)
+()()(U1+U2)(102;0)(B31P1)(EP)
+()()(U1+U2)(304;0)(B32P2)(EP)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(ETOT)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(ETOT)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(ETOT)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(ETOT)
 ETOT   4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(EP)
+()()(U1+U2)(304;0)(B3020)(EP)
+()()(U1+U2)(102;0)(B31P1)(EP)
+()()(U1+U2)(304;0)(B32P2)(EP)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(ETOT)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(ETOT)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(ETOT)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(ETOT)

 M1    1  #  VARIABLE, SET NO., --- [M] * {[M1.NEW] - [M1.OLD]}
 ()()()(;1)(B200)(-M1)
 M1    2  #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
 (-)()(U1+U2)(102;0)(B3010)(M1)
+(-)()(U1+U2)(304;0)(B3020)(M1)
```

```
+(EULER)()()(1;0)(B201)(PRSC)
+(EULER)()()(3;0)(B202)(PRSC)
+(PDUM2,REI)()()(1122;-1)(B211)(U1)
+(PDUM2,REI)()()(3344;-1)(B222)(U1)
+(PDUM2,REI)()()(1324;-1)(B221)(U1)
+(PDUM2,REI)()()(1324;-1)(B212)(U1)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(M1)
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(M1)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(M1)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(M1)
 M1    3  #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
+()()(U1+U2)(102;0)(B3010)(M1)
+()()(U1+U2)(304;0)(B3020)(M1)
+()()(U1+U2)(102;0)(B31P1)(M1)
+()()(U1+U2)(304;0)(B32P2)(M1)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(M1)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(M1)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(M1)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(M1)
 M1    4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(M1)
+()()(U1+U2)(304;0)(B3020)(M1)
+()()(U1+U2)(102;0)(B31P1)(M1)
+()()(U1+U2)(304;0)(B32P2)(M1)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(M1)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(M1)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(M1)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(M1)

 M2    1  #  VARIABLE, SET NO., --- [M] * {[M2.NEW] - [M2.OLD]}
 ()()()(;1)(B200)(-M2)
 M2    2  #  VARIABLE, SET NO., --- {RQ} = [V]{Q} + [D]{Q} + [DB]{Q}
 (-)()(U1+U2)(102;0)(B3010)(M2)
+(-)()(U1+U2)(304;0)(B3020)(M2)
+(EULER)()()(2;0)(B201)(PRSC)
+(EULER)()()(4;0)(B202)(PRSC)
+(PDUM2,REI)()()(1122;-1)(B211)(U2)
+(PDUM2,REI)()()(3344;-1)(B222)(U2)
+(PDUM2,REI)()()(1324;-1)(B221)(U2)
+(PDUM2,REI)()()(1324;-1)(B212)(U2)
+(HBR)()(UMAG,HB1)(1122;-1)(B3011)(M2)
+(HBR)()(UMAG,HB1)(3344;-1)(B3022)(M2)
+(HBR)()(UMAG,HB1)(1324;-1)(B3021)(M2)
+(HBR)()(UMAG,HB1)(1324;-1)(B3012)(M2)
 M2    3  #  VARIABLE, SET NO., ---  INFLOW BOUNDARY SET FOR {RQ}
+()()(U1+U2)(102;0)(B3010)(M2)
+()()(U1+U2)(304;0)(B3020)(M2)
+()()(U1+U2)(102;0)(B31P1)(M2)
+()()(U1+U2)(304;0)(B32P2)(M2)
+(PHRI)()(UMAG,HB1)(1122;-1)(B3011)(M2)
+(PHRI)()(UMAG,HB1)(3344;-1)(B3022)(M2)
+(PHRI)()(UMAG,HB1)(1324;-1)(B3021)(M2)
```

```
+(PHRI)()(UMAG,HB1)(1324;-1)(B3012)(M2)
 M2    4  #  VARIABLE, SET NO., --- OUTFLOW BOUNDARY SET FOR {RQ}
 ()()(U1+U2)(102;0)(B3010)(M2)
+()()(U1+U2)(304;0)(B3020)(M2)
+()()(U1+U2)(102;0)(B31P1)(M2)
+()()(U1+U2)(304;0)(B32P2)(M2)
+(PHRO)()(UMAG,HB1)(1122;-1)(B3011)(M2)
+(PHRO)()(UMAG,HB1)(3344;-1)(B3022)(M2)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3021)(M2)
+(PHRO)()(UMAG,HB1)(1324;-1)(B3012)(M2)
```

JACOBIANS

```
 #
 # FACTORED JACOBIAN FOR DIRECTION 1 #
 #
 RHO RHO 1 1 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 1 #
()()()(;0)(A200)(DETJ)
 RHO RHO 2 1 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 1 #
(-)()(U1+U2)(102;0)(A3010)()
+()()(RHO)(1;0)(A3010)(UOR)+()()(RHO)(2;0)(A3010)(VOR)
+(-,HBJ)()(RHO,HB1)(1122;0)(A3110)(DETC,UMOR)
+(HBJ)()(UMAG,HB1)(1122;0)(A3011)(DETC)
 RHO RHO 3 1 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 1 #
()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(-)()(RHO)(1;0)(A3010)(UOR)+(-)()(RHO)(1;0)(A31P1)(UOR)
+(-)()(RHO)(2;0)(A3010)(VOR)+(-)()(RHO)(2;0)(A31P1)(VOR)
+(-,PHJI)()(RHO,HB1)(1122;0)(A3110)(DETC,UMOR)
+(PHJI)()(UMAG,HB1)(1122;0)(A3011)(DETC)
 RHO RHO 4 1 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 1 #
()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(-)()(RHO)(1;0)(A3010)(UOR)+(-)()(RHO)(1;0)(A31P1)(UOR)
+(-)()(RHO)(2;0)(A3010)(VOR)+(-)()(RHO)(2;0)(A31P1)(VOR)
+(-,PHJO)()(RHO,HB1)(1122;0)(A3110)(DETC,UMOR)
+(PHJO)()(UMAG,HB1)(1122;0)(A3011)(DETC)

 ETOT ETOT 1  1 #
()()()(;0)(A200)(DETJ)
 ETOT ETOT 2  1 #
(-)()(U1+U2)(102;0)(A3010)()
+(-GM1)()(U1+U2)(102;0)(A3010)()
+(PDUM2,PEI)()()(1122;-1)(A211)()
+(HBJ)()(UMAG,HB1)(1122;0)(A3011)(DETC)
 ETOT ETOT 3 1 #
()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(GM1)()(U1+U2)(102;0)(A3010)()+(GM1)()(U1+U2)(102;0)(A31P1)()
+(PHJI)()(UMAG,HB1)(1122;0)(A3011)(DETC)
 ETOT ETOT 4 1 #
()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(GM1)()(U1+U2)(102;0)(A3010)()+(GM1)()(U1+U2)(102;0)(A31P1)()
+(PHJO)()(UMAG,HB1)(1122;0)(A3011)(DETC)

 M1 M1 1  1 #
()()()(;0)(A200)(DETJ)
 M1 M1 2  1 #
(-GM1 )()()(1;0)(A201)(U1)
+(-)()(M1)(1;0)(A3010)(OSRH)
+(-)()(U1+U2)(102;0)(A3010)()
+(PDUM2,REI)()(OSRH)(1122;-1)(A3011)()
+(HBJ)()(M1,HB1)(1122;0)(A3110)(DETC,UORU)
+(HBJ)()(UMAG,HB1)(1122;0)(A3011)(DETC)
M1 M1 3  1 #
()()(M1)(1;0)(A3010)(OSRH)+()()(M1)(1;0)(A31P1)(OSRH)
+()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
```

```
+(PHJI)()(M1,HB1)(1122;0)(A3110)(DETC,UORU)
+(PHJI)()(UMAG,HB1)(1122;0)(A3011)(DETC)
 M1  M1  4   1  #
 (FX2,GM1 )()()(1;0)(A201)(U1)
+()()(M1)(1;0)(A3010)(OSRH)+()()(M1)(1;0)(A31P1)(OSRH)
+()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(PHJO)()(M1,HB1)(1122;0)(A3110)(DETC,UORU)
+(PHJO)()(UMAG,HB1)(1122;0)(A3011)(DETC)


 M2  M2  1   1  #
 ()()()(;0)(A200)(DETJ)
 M2  M2  2   1  #
+(-GM1 )()()(2;0)(A201)(U2)
+(-)()(M2)(2;0)(A3010)(OSRH)
+(-)()(U1+U2)(102;0)(A3010)()
+(PDUM2,REI)()(OSRH)(1122;-1)(A3011)()
+(HBJ)()(M2,HB1)(1122;0)(A3110)(DETC,VORU)
+(HBJ)()(UMAG,HB1)(1122;0)(A3011)(DETC)
 M2  M2  3   1  #
 ()()(M2)(2;0)(A3010)(OSRH)+()()(M2)(2;0)(A31P1)(OSRH)
+()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(PHJI)()(M2,HB1)(1122;0)(A3110)(DETC,VORU)
+(PHJI)()(UMAG,HB1)(1122;0)(A3011)(DETC)
 M2  M2  4   1  #
 (FX2,GM1 )()()(2;0)(A201)(U2)
+()()(M2)(2;0)(A3010)(OSRH)+()()(M2)(2;0)(A31P1)(OSRH)
+()()(U1+U2)(102;0)(A3010)()+()()(U1+U2)(102;0)(A31P1)()
+(PHJO)()(M2,HB1)(1122;0)(A3110)(DETC,VORU)
+(PHJO)()(UMAG,HB1)(1122;0)(A3011)(DETC)

 RHO  M1  2   1  #
(-)()(RHO)(1;0)(A3010)(OSRH)
+(HBJ)()(RHO,HB1)(1122;0)(A3110)(DETC,UORU)
 RHO  M1  3   1  #
 ()()(RHO)(1;0)(A3010)(OSRH)+()()(RHO)(1;0)(A31P1)(OSRH)
+(PHJI)()(RHO,HB1)(1122;0)(A3110)(DETC,UORU)
 RHO  M1  4   1  #
 ()()(RHO)(1;0)(A3010)(OSRH)+()()(RHO)(1;0)(A31P1)(OSRH)
+(PHJO)()(RHO,HB1)(1122;0)(A3110)(DETC,UORU)

 RHO  M2  2   1
(-)()(RHO)(2;0)(A3010)(OSRH)
+(HBJ)()(RHO,HB1)(1122;0)(A3110)(DETC,VORU)
 RHO  M2  3   1  #
 ()()(RHO)(2;0)(A3010)(OSRH)+()()(RHO)(2;0)(A31P1)(OSRH)
+(PHJI)()(RHO,HB1)(1122;0)(A3110)(DETC,VORU)
 RHO  M2  4   1  #
 ()()(RHO)(2;0)(A3010)(OSRH)+()()(RHO)(2;0)(A31P1)(OSRH)
+(PHJO)()(RHO,HB1)(1122;0)(A3110)(DETC,VORU)

 ETOT  RHO  2   1  #
```

```
()()(EP)(1;0)(A3010)(UOR)+()()(EP)(2;0)(A3010)(VOR)
+(HGMMKI)()(U1+U2)(102;0)(A3010)(UVWS)
+(HBJ)()(ETOT,HB1)(1122;0)(A3110)(DETC,UMOR)
 ETOT RHO 3 1 #
(-)()(EP)(1;0)(A3010)(UOR)+(-)()(EP)(1;0)(A31P1)(UOR)
+(-)()(EP)(2;0)(A3010)(VOR)+(-)()(EP)(2;0)(A31P1)(VOR)
+(-,HGMMKI)()(U1+U2)(102;0)(A3010)(UVWS)
+(-,HGMMKI)()(U1+U2)(102;0)(A31P1)(UVWS)
+(PHJI)()(ETOT,HB1)(1122;0)(A3110)(DETC,UMOR)
 ETOT RHO 4 1 #
(-)()(EP)(1;0)(A3010)(UOR)+(-)()(EP)(1;0)(A31P1)(UOR)
+(-)()(EP)(2;0)(A3010)(VOR)+(-)()(EP)(2;0)(A31P1)(VOR)
+(-,HGMMKI)()(U1+U2)(102;0)(A3010)(UVWS)
+(NF2,-,HGMMKI)()(U1+U2)(102;0)(A31P1)(UVWS)
+(PHJO)()(ETOT,HB1)(1122;0)(A3110)(DETC,UMOR)


 ETOT M1 2 1 #
(-)()(EP)(1;0)(A3010)(OSRH)
+(GMMKI)()(U1+U2)(102;0)(A3010)(U1)
+(HBJ)()(ETOT,HB1)(1122;0)(A3110)(DETC,UORU)
 ETOT M1 3 1 #
()()(EP)(1;0)(A3010)(OSRH)+()()(EP)(1;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(A3010)(U1)
+(-GMMKI)()(U1+U2)(102;0)(A31P1)(U1)
+(PHJI)()(ETOT,HB1)(1122;0)(A3110)(DETC,UORU)
 ETOT M1 4 1 #
()()(EP)(1;0)(A3010)(OSRH)+()()(EP)(1;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(A3010)(U1)
+(NF2,-GMMKI)()(U1+U2)(102;0)(A31P1)(U1)
+(PHJO)()(ETOT,HB1)(1122;0)(A3110)(DETC,UORU)

 ETOT M2 2 1 #
(-)()(EP)(2;0)(A3010)(OSRH)
+(GMMKI)()(U1+U2)(102;0)(A3010)(U2)
+(HBJ)()(ETOT,HB1)(1122;0)(A3110)(DETC,VORU)
 ETOT M2 3 1 #
()()(EP)(2;0)(A3010)(OSRH)+()()(EP)(2;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(A3010)(U2)
+(-GMMKI)()(U1+U2)(102;0)(A31P1)(U2)
+(PHJI)()(ETOT,HB1)(1122;0)(A3110)(DETC,VORU)
 ETOT M2 4 1 #
()()(EP)(2;0)(A3010)(OSRH)+()()(EP)(2;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(102;0)(A3010)(U2)
+(NF2,-GMMKI)()(U1+U2)(102;0)(A31P1)(U2)
+(PHJO)()(ETOT,HB1)(1122;0)(A3110)(DETC,VORU)

 M1 RHO 2 1 #
(GMH)()()(1;0)(A201)(UVWS)
+()()(M1)(1;0)(A3010)(UOR)+()()(M1)(2;0)(A3010)(VOR)
+(HBJ)()(M1,HB1)(1122;0)(A3110)(DETC,UMOR)
 M1 RHO 3 1 #
```

```
 (-)()(M1)(1;0)(A3010)(UOR)+(-)()(M1)(1;0)(A31P1)(UOR)
+(-)()(M1)(2;0)(A3010)(VOR)+(-)()(M1)(2;0)(A31P1)(VOR)
+(PHJI)()(M1,HB1)(1122;0)(A3110)(DETC,UMOR)
 M1 RHO 4  1 #
 (FX2,-GMH)()()(1;0)(A201)(UVWS)
+(-)()(M1)(1;0)(A3010)(UOR)+(-)()(M1)(1;0)(A31P1)(UOR)
+(-)()(M1)(2;0)(A3010)(VOR)+(-)()(M1)(2;0)(A31P1)(VOR)
+(PHJO)()(M1,HB1)(1122;0)(A3110)(DETC,UMOR)

 M1 ETOT 2  1 #
 (GM1MK)()()(1;0)(A201)()
 M1 ETOT 4  1 #
 (FX2,-GM1MK)()()(1;0)(A201)()

 M1 M2 2  1 #
 (-GM1  )()()(1;0)(A201)(U2)
+(-)()(M1)(2;0)(A3010)(OSRH)
+(HBJ)()(M1,HB1)(1122;0)(A3110)(DETC,VORU)
 M1 M2 3  1 #
 ()()(M1)(2;0)(A3010)(OSRH)+()()(M1)(2;0)(A31P1)(OSRH)
+(PHJI)()(M1,HB1)(1122;0)(A3110)(DETC,VORU)
 M1 M2 4  1 #
 (FX2,GM1  )()()(1;0)(A201)(U2)
+()()(M1)(2;0)(A3010)(OSRH)+()()(M1)(2;0)(A31P1)(OSRH)
+(PHJO)()(M1,HB1)(1122;0)(A3110)(DETC,VORU)

 M2 RHO 2 1 #
 (GMH)()()(2;0)(A201)(UVWS)
+()()(M2)(1;0)(A3010)(UOR)+()()(M2)(2;0)(A3010)(VOR)
+(HBJ)()(M2,HB1)(1122;0)(A3110)(DETC,UMOR)
 M2 RHO 3 1 #
 (-)()(M2)(1;0)(A3010)(UOR)+(-)()(M2)(1;0)(A31P1)(UOR)
+(-)()(M2)(2;0)(A3010)(VOR)+(-)()(M2)(2;0)(A31P1)(VOR)
+(PHJI)()(M2,HB1)(1122;0)(A3110)(DETC,UMOR)
 M2 RHO 4 1 #
 (FX2,-GMH)()()(2;0)(A201)(UVWS)
+(-)()(M2)(1;0)(A3010)(UOR)+(-)()(M2)(1;0)(A31P1)(UOR)
+(-)()(M2)(2;0)(A3010)(VOR)+(-)()(M2)(2;0)(A31P1)(VOR)
+(PHJO)()(M2,HB1)(1122;0)(A3110)(DETC,UMOR)

 M2 ETOT 2  1 #
 (GM1MK)()()(2;0)(A201)()
 M2 ETOT 4  1 #
 (FX2,-GM1MK)()()(2;0)(A201)()

 M2 M1 2  1 #
 (-GM1  )()()(2;0)(A201)(U1)
+(-)()(M2)(1;0)(A3010)(OSRH)
+(HBJ)()(M2,HB1)(1122;0)(A3110)(DETC,UORU)
 M2 M1 3  1 #
 ()()(M2)(1;0)(A3010)(OSRH)+()()(M2)(1;0)(A31P1)(OSRH)
+(PHJI)()(M2,HB1)(1122;0)(A3110)(DETC,UORU)
```

```
 M2  M1  4   1  #
 (FX2,GM1   )()()(2;0)(A201)(U1)
+()()(M2)(1;0)(A3010)(OSRH)+()()(M2)(1;0)(A31P1)(OSRH)
+(PHJO)()(M2,HB1)(1122;0)(A3110)(DETC,UORU)
```

JACOBIANS

```
 #
 # FACTORED JACOBIAN FOR DIRECTION 2 #
 #

 RHO RHO 1 2 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2 #
()()()(;0)(A200)(DETJ)
 RHO RHO 2 2 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2 #
(-)()(U1+U2)(304;0)(A3010)()
+()()(RHO)(3;0)(A3010)(UOR)+()()(RHO)(4;0)(A3010)(VOR)
+(-,HBJ)()(RHO,HB1)(3344;0)(A3110)(DETC,UMOR)
+(HBJ)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 RHO RHO 3 2 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2 #
()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(-)()(RHO)(3;0)(A3010)(UOR)+(-)()(RHO)(3;0)(A31P1)(UOR)
+(-)()(RHO)(4;0)(A3010)(VOR)+(-)()(RHO)(4;0)(A31P1)(VOR)
+(-,PHJI)()(RHO,HB1)(3344;0)(A3110)(DETC,UMOR)
+(PHJI)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 RHO RHO 4 2 # VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2 #
()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(-)()(RHO)(3;0)(A3010)(UOR)+(-)()(RHO)(3;0)(A31P1)(UOR)
+(-)()(RHO)(4;0)(A3010)(VOR)+(-)()(RHO)(4;0)(A31P1)(VOR)
+(-,PHJO)()(RHO,HB1)(3344;0)(A3110)(DETC,UMOR)
+(PHJO)()(UMAG,HB1)(3344;0)(A3011)(DETC)

 ETOT ETOT 1 2 #
()()()(;0)(A200)(DETJ)
 ETOT ETOT 2 2 #
(-)()(U1+U2)(304;0)(A3010)()
+(-GM1)()(U1+U2)(304;0)(A3010)()
+(PDUM2,PEI)()()(3344;-1)(A211)()
+(HBJ)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 ETOT ETOT 3 2 #
()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(GM1)()(U1+U2)(304;0)(A3010)()+(GM1)()(U1+U2)(304;0)(A31P1)()
+(PHJI)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 ETOT ETOT 4 2 #
()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(GM1)()(U1+U2)(304;0)(A3010)()+(GM1)()(U1+U2)(304;0)(A31P1)()
+(PHJO)()(UMAG,HB1)(3344;0)(A3011)(DETC)

 M1 M1 1 2 #
()()()(;0)(A200)(DETJ)
 M1 M1 2 2 #
+(-GM1 )()()(3;0)(A201)(U1)
+(-)()(M1)(3;0)(A3010)(OSRH)
+(-)()(U1+U2)(304;0)(A3010)()
+(PDUM2,REI)()(OSRH)(3344;-1)(A3011)()
+(HBJ)()(M1,HB1)(3344;0)(A3110)(DETC,UORU)
+(HBJ)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 M1 M1 3 2 #
()()(M1)(3;0)(A3010)(OSRH)+()()(M1)(3;0)(A31P1)(OSRH)
```

```
+()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(PHJI)()(M1,HB1)(3344;0)(A3110)(DETC,UORU)
+(PHJI)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 M1  M1  4   2  #
 (FX2,GM1 )()()(3;0)(A201)(U1)
+()()(M1)(3;0)(A3010)(OSRH)+()()(M1)(3;0)(A31P1)(OSRH)
+()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(PHJO)()(M1,HB1)(3344;0)(A3110)(DETC,UORU)
+(PHJO)()(UMAG,HB1)(3344;0)(A3011)(DETC)


 M2  M2 1   2  #
 ()()()(;0)(A200)(DETJ)
 M2  M2 2   2  #
+(-GM1 )()()(4;0)(A201)(U2)
+(-)()(M2)(4;0)(A3010)(OSRH)
+(-)()(U1+U2)(304;0)(A3010)()
+(PDUM2,REI)()(OSRH)(3344;-1)(A3011)()
+(HBJ)()(M2,HB1)(3344;0)(A3110)(DETC,VORU)
+(HBJ)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 M2  M2  3   2  #
 ()()(M2)(4;0)(A3010)(OSRH)+()()(M2)(4;0)(A31P1)(OSRH)
+()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(PHJI)()(M2,HB1)(3344;0)(A3110)(DETC,VORU)
+(PHJI)()(UMAG,HB1)(3344;0)(A3011)(DETC)
 M2  M2  4   2  #
 (FX2,GM1 )()()(4;0)(A201)(U2)
+()()(M2)(4;0)(A3010)(OSRH)+()()(M2)(4;0)(A31P1)(OSRH)
+()()(U1+U2)(304;0)(A3010)()+()()(U1+U2)(304;0)(A31P1)()
+(PHJO)()(M2,HB1)(3344;0)(A3110)(DETC,VORU)
+(PHJO)()(UMAG,HB1)(3344;0)(A3011)(DETC)

 RHO  M1  2   2  #
 (-)()(RHO)(3;0)(A3010)(OSRH)
+(HBJ)()(RHO,HB1)(3344;0)(A3110)(DETC,UORU)
 RHO  M1  3   2  #
 ()()(RHO)(3;0)(A3010)(OSRH)+()()(RHO)(3;0)(A31P1)(OSRH)
+(PHJI)()(RHO,HB1)(3344;0)(A3110)(DETC,UORU)
 RHO  M1  4   2  #
 ()()(RHO)(3;0)(A3010)(OSRH)+()()(RHO)(3;0)(A31P1)(OSRH)
+(PHJO)()(RHO,HB1)(3344;0)(A3110)(DETC,UORU)

 RHO  M2  2   2  #
 (-)()(RHO)(4;0)(A3010)(OSRH)
+(HBJ)()(RHO,HB1)(3344;0)(A3110)(DETC,VORU)
 RHO  M2  3   2  #
 ()()(RHO)(4;0)(A3010)(OSRH)+()()(RHO)(4;0)(A31P1)(OSRH)
+(PHJI)()(RHO,HB1)(3344;0)(A3110)(DETC,VORU)
 RHO  M2  4   2  #
 ()()(RHO)(4;0)(A3010)(OSRH)+()()(RHO)(4;0)(A31P1)(OSRH)
+(PHJO)()(RHO,HB1)(3344;0)(A3110)(DETC,VORU)
```

```
 ETOT RHO 2  2 #
()()(EP)(3;0)(A3010)(UOR)+()()(EP)(4;0)(A3010)(VOR)
+(HGMMKI)()(U1+U2)(304;0)(A3010)(UVWS)
+(HBJ)()(ETOT,HB1)(3344;0)(A3110)(DETC,UMOR)
 ETOT RHO 3  2 #
(-)()(EP)(3;0)(A3010)(UOR)+(-)()(EP)(3;0)(A31P1)(UOR)
+(-)()(EP)(4;0)(A3010)(VOR)+(-)()(EP)(4;0)(A31P1)(VOR)
+(-,HGMMKI)()(U1+U2)(304;0)(A3010)(UVWS)
+(-,HGMMKI)()(U1+U2)(304;0)(A31P1)(UVWS)
+(PHJI)()(ETOT,HB1)(3344;0)(A3110)(DETC,UMOR)
 ETOT RHO 4  2 #
(-)()(EP)(3;0)(A3010)(UOR)+(-)()(EP)(3;0)(A31P1)(UOR)
+(-)()(EP)(4;0)(A3010)(VOR)+(-)()(EP)(4;0)(A31P1)(VOR)
+(-,HGMMKI)()(U1+U2)(304;0)(A3010)(UVWS)
+(NF2,-,HGMMKI)()(U1+U2)(304;0)(A31P1)(UVWS)
+(PHJO)()(ETOT,HB1)(3344;0)(A3110)(DETC,UMOR)


 ETOT M1 2  2 #
(-)()(EP)(3;0)(A3010)(OSRH)
+(GMMKI)()(U1+U2)(304;0)(A3010)(U1)
+(HBJ)()(ETOT,HB1)(3344;0)(A3110)(DETC,UORU)
 ETOT M1 3  2 #
()()(EP)(3;0)(A3010)(OSRH)+()()(EP)(3;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(304;0)(A3010)(U1)
+(-GMMKI)()(U1+U2)(304;0)(A31P1)(U1)
+(PHJI)()(ETOT,HB1)(3344;0)(A3110)(DETC,UORU)
 ETOT M1 4  2 #
()()(EP)(3;0)(A3010)(OSRH)+()()(EP)(3;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(304;0)(A3010)(U1)
+(NF2,-GMMKI)()(U1+U2)(304;0)(A31P1)(U1)
+(PHJO)()(ETOT,HB1)(3344;0)(A3110)(DETC,UORU)

 ETOT M2 2  2 #
(-)()(EP)(4;0)(A3010)(OSRH)
+(GMMKI)()(U1+U2)(304;0)(A3010)(U2)
+(HBJ)()(ETOT,HB1)(3344;0)(A3110)(DETC,VORU)
 ETOT M2 3  2 #
()()(EP)(4;0)(A3010)(OSRH)+()()(EP)(4;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(304;0)(A3010)(U2)
+(-GMMKI)()(U1+U2)(304;0)(A31P1)(U2)
+(PHJI)()(ETOT,HB1)(3344;0)(A3110)(DETC,VORU)
 ETOT M2 4  2 #
()()(EP)(4;0)(A3010)(OSRH)+()()(EP)(4;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2)(304;0)(A3010)(U2)
+(NF2,-GMMKI)()(U1+U2)(304;0)(A31P1)(U2)
+(PHJO)()(ETOT,HB1)(3344;0)(A3110)(DETC,VORU)

 M1 RHO 2  2 #
(GMH)()()(3;0)(A201)(UVWS)
+()()(M1)(3;0)(A3010)(UOR)+()()(M1)(4;0)(A3010)(VOR)
+(HBJ)()(M1,HB1)(3344;0)(A3110)(DETC,UMOR)
```

```
 M1  RHO  3   2  #
(-)()(M1)(3;0)(A3010)(UOR)+(-)()(M1)(3;0)(A31P1)(UOR)
+(-)()(M1)(4;0)(A3010)(VOR)+(-)()(M1)(4;0)(A31P1)(VOR)
+(PHJI)()(M1,HB1)(3344;0)(A3110)(DETC,UMOR)
 M1  RHO  4   2  #
(FX2,-GMH)()()(3;0)(A201)(UVWS)
+(-)()(M1)(3;0)(A3010)(UOR)+(-)()(M1)(3;0)(A31P1)(UOR)
+(-)()(M1)(4;0)(A3010)(VOR)+(-)()(M1)(4;0)(A31P1)(VOR)
+(PHJO)()(M1,HB1)(3344;0)(A3110)(DETC,UMOR)

 M1  ETOT  2   2  #
(GM1MK)()()(3;0)(A201)()
 M1  ETOT  4   2  #
(FX2,-GM1MK)()()(3;0)(A201)()

 M1  M2  2   2  #
(-GM1   )()()(3;0)(A201)(U2)
+(-)()(M1)(4;0)(A3010)(OSRH)
+(HBJ)()(M1,HB1)(3344;0)(A3110)(DETC,VORU)
 M1  M2  3   2  #
()()(M1)(4;0)(A3010)(OSRH)+()()(M1)(4;0)(A31P1)(OSRH)
+(PHJI)()(M1,HB1)(3344;0)(A3110)(DETC,VORU)
 M1  M2  4   2  #
(FX2,GM1   )()()(3;0)(A201)(U2)
+()()(M1)(4;0)(A3010)(OSRH)+()()(M1)(4;0)(A31P1)(OSRH)
+(PHJO)()(M1,HB1)(3344;0)(A3110)(DETC,VORU)

 M2  RHO  2   2  #
(GMH)()()(4;0)(A201)(UVWS)
+()()(M2)(3;0)(A3010)(UOR)+()()(M2)(4;0)(A3010)(VOR)
+(HBJ)()(M2,HB1)(3344;0)(A3110)(DETC,UMOR)
 M2  RHO  3   2  #
(-)()(M2)(3;0)(A3010)(UOR)+(-)()(M2)(3;0)(A31P1)(UOR)
+(-)()(M2)(4;0)(A3010)(VOR)+(-)()(M2)(4;0)(A31P1)(VOR)
+(PHJI)()(M2,HB1)(3344;0)(A3110)(DETC,UMOR)
 M2  RHO  4   2  #
(FX2,-GMH)()()(4;0)(A201)(UVWS)
+(-)()(M2)(3;0)(A3010)(UOR)+(-)()(M2)(3;0)(A31P1)(UOR)
+(-)()(M2)(4;0)(A3010)(VOR)+(-)()(M2)(4;0)(A31P1)(VOR)
+(PHJO)()(M2,HB1)(3344;0)(A3110)(DETC,UMOR)

 M2  ETOT  2   2  #
(GM1MK)()()(4;0)(A201)()
 M2  ETOT  4   2  #
(FX2,-GM1MK)()()(4;0)(A201)()

 M2  M1  2   2  #
(-GM1   )()()(4;0)(A201)(U1)
+(-)()(M2)(3;0)(A3010)(OSRH)
+(HBJ)()(M2,HB1)(3344;0)(A3110)(DETC,UORU)
 M2  M1  3   2  #
()()(M2)(3;0)(A3010)(OSRH)+()()(M2)(3;0)(A31P1)(OSRH)
```

```
+(PHJI)()(M2,HB1)(3344;0)(A3110)(DETC,UORU)
 M2  M1  4   2  #
 (FX2,GM1   )()()(4;0)(A201)(U1)
+()()(M2)(3;0)(A3010)(OSRH)+()()(M2)(3;0)(A31P1)(OSRH)
+(PHJO)()(M2,HB1)(3344;0)(A3110)(DETC,UORU)

GROUP FREQUENCY
   1
SOLUTION TYPE
 DELTA_Q
 FACTORED_GAUSS_ELIMINATION
 IMPLICIT_EULER

END
```

APPENDIX D
AKCESS.AERO *REMI* template, quasi-Newton TP, *d*=3

```
INTEGRATION FACTORS
  INITIAL_TIME
  FINAL_TIME
  PROBLEM_CONVERGENCE_CRITERIA
  MAXIMUM_CHANGE_IN_Q__(DQ)
  INITIAL_TIME_STEP
  TIME_STEP_MULTIPLIER
  MAXIMUM_TIME_STEP
  CRITERIA_TO_RAISE_MAX_TIME_STEP
  MAXIMUM_NUMBER_OF_STEPS
  MAXIMUM_NUMBER_OF_ITERATIONS_PER_STEP
  ITERATION_CONVERGENCE_CRITERIA
  THETA__IMPLICITNESS_FACTOR
  MAXIMUM_VALUE_OF_ANY_DELTA_Q

TRANSFORMATION ARRAYS
    ETKJ    1.
    DETJ    1.
#    DETE    0.

BOUNDARY CONDITIONS
        RHO ETOT M1 M2 M3 PRSC     # ORDER
 DIR_RT    D   D 0 0 0   0   # DIRICHLET
 DIR_M1    0   0 D 0 0   0 # DIRICHLET
 DIR_M2    0   0 0 D 0   0 # DIRICHLET
 DIR_M3    0   0 0 0 D   0 # DIRICHLET
 DIR_PRS   0   0 0 0 0   D   # DIRICHLET
 THR_IN  [3 -3] [3 -3] [3 -3] [3 -3] [3 -3]  0    # THROUGHFLOW
 THR_OUT  [4 -4] [4 -4] [4 -4] [4 -4] [4 -4]  0    # THROUGHFLOW

TITLE          **** TEMPLATE FILE TEMP.CNS2D.REMI ****
 CNS3D  ALLMACH ALGORITHM,  TENSOR MATRIX A JACOBIANS (2/23/93)

RESIDUALS

 RHO  1  #  VARBL, SET NO., # OF TERMS --- TEMPORAL SET (RHO)
 ()()()(;1)(C200)(-RHO)
 RHO  2  #  VARBL, SET NO., # OF TERMS --- SPATIAL  SET (RHO)
 (-)()(U1+U2+U3)(10203;0)(C3010)(RHO)+(-)()(U1+U2+U3)(40506;0)(C3020)(RHO)
+(-)()(U1+U2+U3)(70809;0)(C3030)(RHO)
+(HBR)()(UMHB)(112233;-1)(C3011)(RHO)
+(HBR)()(UMHB)(445566;-1)(C3022)(RHO)
+(HBR)()(UMHB)(778899;-1)(C3033)(RHO)
+(HBR)()(UMHB)(142536;-1)(C3021)(RHO)
+(HBR)()(UMHB)(142536;-1)(C3012)(RHO)
+(HBR)()(UMHB)(172839;-1)(C3031)(RHO)
+(HBR)()(UMHB)(172839;-1)(C3013)(RHO)
+(HBR)()(UMHB)(475869;-1)(C3023)(RHO)
+(HBR)()(UMHB)(475869;-1)(C3032)(RHO)
```

```
 RHO  3  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (RHO)
 ()()(U1+U2+U3)(10203;0)(C3010)(RHO)+()()(U1+U2+U3)(40506;0)(C3020)(RHO)
+()()(U1+U2+U3)(70809;0)(C3030)(RHO)
+()()(U1+U2+U3)(10203;0)(C31P1)(RHO)+()()(U1+U2+U3)(40506;0)(C32P2)(RHO)
+()()(U1+U2+U3)(70809;0)(C33P3)(RHO)
 RHO  4  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (RHO)
 ()()(U1+U2+U3)(10203;0)(C3010)(RHO)+()()(U1+U2+U3)(40506;0)(C3020)(RHO)
+()()(U1+U2+U3)(70809;0)(C3030)(RHO)
+()()(U1+U2+U3)(10203;0)(C31P1)(RHO)+()()(U1+U2+U3)(40506;0)(C32P2)(RHO)
+()()(U1+U2+U3)(70809;0)(C33P3)(RHO)  ETOT  1  #  VARBL, SET NO., # OF TERMS --- TEMPORAL
SET (ETOT)
 ()()()(;1)(C200)(-ETOT)
 ETOT  2  # VARBL, SET NO., # OF TERMS --- SPATIAL  SET (ETOT)
 (-)()(U1+U2+U3)(10203;0)(C3010)(EP)+(-)()(U1+U2+U3)(40506;0)(C3020)(EP)
+(-)()(U1+U2+U3)(70809;0)(C3030)(EP)
+(ZPEC,PEI)()()(112233;-1)(C211)(TEMP)
+(ZPEC,PEI)()()(445566;-1)(C222)(TEMP)
+(ZPEC,PEI)()()(778899;-1)(C233)(TEMP)
+(ZPEC,PEI)()()(142536;-1)(C221)(TEMP)
+(ZPEC,PEI)()()(142536;-1)(C212)(TEMP)
+(ZPEC,PEI)()()(172839;-1)(C231)(TEMP)
+(ZPEC,PEI)()()(172839;-1)(C213)(TEMP)
+(ZPEC,PEI)()()(475869;-1)(C223)(TEMP)
+(ZPEC,PEI)()()(475869;-1)(C232)(TEMP)
+(HBR)()(UMHB)(112233;-1)(C3011)(ETOT)
+(HBR)()(UMHB)(445566;-1)(C3022)(ETOT)
+(HBR)()(UMHB)(778899;-1)(C3033)(ETOT)
+(HBR)()(UMHB)(142536;-1)(C3021)(ETOT)
+(HBR)()(UMHB)(142536;-1)(C3012)(ETOT)
+(HBR)()(UMHB)(172839;-1)(C3031)(ETOT)
+(HBR)()(UMHB)(172839;-1)(C3013)(ETOT)
+(HBR)()(UMHB)(475869;-1)(C3023)(ETOT)
+(HBR)()(UMHB)(475869;-1)(C3032)(ETOT)
 ETOT  3  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (ETOT)
 ()()(U1+U2+U3)(10203;0)(C3010)(EP)+()()(U1+U2+U3)(40506;0)(C3020)(EP)
+()()(U1+U2+U3)(70809;0)(C3030)(EP)
+()()(U1+U2+U3)(10203;0)(C31P1)(EP)+()()(U1+U2+U3)(40506;0)(C32P2)(EP)
+()()(U1+U2+U3)(70809;0)(C33P3)(EP)
 ETOT  4  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (ETOT)
 ()()(U1+U2+U3)(10203;0)(C3010)(EP)+()()(U1+U2+U3)(40506;0)(C3020)(EP)
+()()(U1+U2+U3)(70809;0)(C3030)(EP)
+()()(U1+U2+U3)(10203;0)(C31P1)(EP)+()()(U1+U2+U3)(40506;0)(C32P2)(EP)
+()()(U1+U2+U3)(70809;0)(C33P3)(EP)

 M1  1  #  VARBL, SET NO., # OF TERMS --- TEMPORAL SET (M1)
 ()()()(;1)(C200)(-M1)
 M1  2  #  VARBL, SET NO., # OF TERMS --- SPATIAL  SET (M1)
 (-)()(U1+U2+U3)(10203;0)(C3010)(M1)+(-)()(U1+U2+U3)(40506;0)(C3020)(M1)
+(-)()(U1+U2+U3)(70809;0)(C3030)(M1)
+(EULER)()()(1;0)(C201)(PRSC)+(EULER)()()(4;0)(C202)(PRSC)
+(EULER)()()(7;0)(C203)(PRSC)
+(PDUM2,REI)()()(112233;-1)(C211)(U1)
```

```
+(PDUM2,REI)()()(445566;-1)(C222)(U1)
+(PDUM2,REI)()()(778899;-1)(C233)(U1)
+(PDUM2,REI)()()(142536;-1)(C221)(U1)
+(PDUM2,REI)()()(142536;-1)(C212)(U1)
+(PDUM2,REI)()()(172839;-1)(C231)(U1)
+(PDUM2,REI)()()(172839;-1)(C213)(U1)
+(PDUM2,REI)()()(475869;-1)(C223)(U1)
+(PDUM2,REI)()()(475869;-1)(C232)(U1)
+(HBR)()(UMHB)(112233;-1)(C3011)(M1)
+(HBR)()(UMHB)(445566;-1)(C3022)(M1)
+(HBR)()(UMHB)(778899;-1)(C3033)(M1)
+(HBR)()(UMHB)(142536;-1)(C3021)(M1)
+(HBR)()(UMHB)(142536;-1)(C3012)(M1)
+(HBR)()(UMHB)(172839;-1)(C3031)(M1)
+(HBR)()(UMHB)(172839;-1)(C3013)(M1)
+(HBR)()(UMHB)(475869;-1)(C3023)(M1)
+(HBR)()(UMHB)(475869;-1)(C3032)(M1)
 M1  3  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (M1)
 ()()(U1+U2+U3)(10203;0)(C3010)(M1)+()()(U1+U2+U3)(40506;0)(C3020)(M1)
+()()(U1+U2+U3)(70809;0)(C3030)(M1)
+()()(U1+U2+U3)(10203;0)(C31P1)(M1)+()()(U1+U2+U3)(40506;0)(C32P2)(M1)
+()()(U1+U2+U3)(70809;0)(C33P3)(M1)
 M1  4  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (M1)
 ()()(U1+U2+U3)(10203;0)(C3010)(M1)+()()(U1+U2+U3)(40506;0)(C3020)(M1)
+()()(U1+U2+U3)(70809;0)(C3030)(M1)
+()()(U1+U2+U3)(10203;0)(C31P1)(M1)+()()(U1+U2+U3)(40506;0)(C32P2)(M1)
+()()(U1+U2+U3)(70809;0)(C33P3)(M1)
 M2  1  #  VARBL, SET NO., # OF TERMS --- TEMPORAL SET (M2)
 ()()()(;1)(C200)(-M2)
 M2  2  #  VARBL, SET NO., # OF TERMS --- SPATIAL  SET (M2)
 (-)()(U1+U2+U3)(10203;0)(C3010)(M2)+(-)()(U1+U2+U3)(40506;0)(C3020)(M2)
+(-)()(U1+U2+U3)(70809;0)(C3030)(M2)
+(EULER)()()(2;0)(C201)(PRSC)+(EULER)()()(5;0)(C202)(PRSC)
+(EULER)()()(8;0)(C203)(PRSC)
+(PDUM2,REI)()()(112233;-1)(C211)(U2)
+(PDUM2,REI)()()(445566;-1)(C222)(U2)
+(PDUM2,REI)()()(778899;-1)(C233)(U2)
+(PDUM2,REI)()()(142536;-1)(C221)(U2)
+(PDUM2,REI)()()(142536;-1)(C212)(U2)
+(PDUM2,REI)()()(172839;-1)(C231)(U2)
+(PDUM2,REI)()()(172839;-1)(C213)(U2)
+(PDUM2,REI)()()(475869;-1)(C223)(U2)
+(PDUM2,REI)()()(475869;-1)(C232)(U2)
+(HBR)()(UMHB)(112233;-1)(C3011)(M2)
+(HBR)()(UMHB)(445566;-1)(C3022)(M2)
+(HBR)()(UMHB)(778899;-1)(C3033)(M2)
+(HBR)()(UMHB)(142536;-1)(C3021)(M2)
+(HBR)()(UMHB)(142536;-1)(C3012)(M2)
+(HBR)()(UMHB)(172839;-1)(C3031)(M2)
+(HBR)()(UMHB)(172839;-1)(C3013)(M2)
+(HBR)()(UMHB)(475869;-1)(C3023)(M2)
+(HBR)()(UMHB)(475869;-1)(C3032)(M2)
```

```
 M2  3  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (M2)
 ()()(U1+U2+U3)(10203;0)(C3010)(M2)+()()(U1+U2+U3)(40506;0)(C3020)(M2)
+()()(U1+U2+U3)(70809;0)(C3030)(M2)
+()()(U1+U2+U3)(10203;0)(C31P1)(M2)+()()(U1+U2+U3)(40506;0)(C32P2)(M2)
+()()(U1+U2+U3)(70809;0)(C33P3)(M2)
 M2  4  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (M2)
 ()()(U1+U2+U3)(10203;0)(C3010)(M2)+()()(U1+U2+U3)(40506;0)(C3020)(M2)
+()()(U1+U2+U3)(70809;0)(C3030)(M2)
+()()(U1+U2+U3)(10203;0)(C31P1)(M2)+()()(U1+U2+U3)(40506;0)(C32P2)(M2)
+()()(U1+U2+U3)(70809;0)(C33P3)(M2)
 M3  1  #  VARBL, SET NO., # OF TERMS --- TEMPORAL SET (M3)
 ()()()(;1)(C200)(-M3)
 M3  2  #  VARBL, SET NO., # OF TERMS --- SPATIAL  SET (M3)
 (-)()(U1+U2+U3)(10203;0)(C3010)(M3)+(-)()(U1+U2+U3)(40506;0)(C3020)(M3)
+(-)()(U1+U2+U3)(70809;0)(C3030)(M3)
+(EULER)()()(3;0)(C201)(PRSC)+(EULER)()()(6;0)(C202)(PRSC)
+(EULER)()()(9;0)(C203)(PRSC)
+(PDUM2,REI)()()(112233;-1)(C211)(U3)
+(PDUM2,REI)()()(445566;-1)(C222)(U3)
+(PDUM2,REI)()()(778899;-1)(C233)(U3)
+(PDUM2,REI)()()(142536;-1)(C221)(U3)
+(PDUM2,REI)()()(142536;-1)(C212)(U3)
+(PDUM2,REI)()()(172839;-1)(C231)(U3)
+(PDUM2,REI)()()(172839;-1)(C213)(U3)
+(PDUM2,REI)()()(475869;-1)(C223)(U3)
+(PDUM2,REI)()()(475869;-1)(C232)(U3)
+(HBR)()(UMHB)(112233;-1)(C3011)(M3)
+(HBR)()(UMHB)(445566;-1)(C3022)(M3)
+(HBR)()(UMHB)(778899;-1)(C3033)(M3)
+(HBR)()(UMHB)(142536;-1)(C3021)(M3)
+(HBR)()(UMHB)(142536;-1)(C3012)(M3)
+(HBR)()(UMHB)(172839;-1)(C3031)(M3)
+(HBR)()(UMHB)(172839;-1)(C3013)(M3)
+(HBR)()(UMHB)(475869;-1)(C3023)(M3)
+(HBR)()(UMHB)(475869;-1)(C3032)(M3)
 M3  3  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (M3)
 ()()(U1+U2+U3)(10203;0)(C3010)(M3)+()()(U1+U2+U3)(40506;0)(C3020)(M3)
+()()(U1+U2+U3)(70809;0)(C3030)(M3)
+()()(U1+U2+U3)(10203;0)(C31P1)(M3)+()()(U1+U2+U3)(40506;0)(C32P2)(M3)
+()()(U1+U2+U3)(70809;0)(C33P3)(M3)
 M3  4  #  VARBL, SET NO., # OF TERMS --- BOUNDARY SET (M3)
 ()()(U1+U2+U3)(10203;0)(C3010)(M3)+()()(U1+U2+U3)(40506;0)(C3020)(M3)
+()()(U1+U2+U3)(70809;0)(C3030)(M3)
+()()(U1+U2+U3)(10203;0)(C31P1)(M3)+()()(U1+U2+U3)(40506;0)(C32P2)(M3)
+()()(U1+U2+U3)(70809;0)(C33P3)(M3)
```

JACOBIANS

```
 #
 # FACTORED JACOBIAN FOR DIRECTION 1 #
 #
 RHO RHO  1  1 # VARBL, VARDIF, SET,  DIRECTION 1 #
()()()(;0)(A200)(DETJ)
 RHO RHO  2  1 # VARBL, VARDIF, SET,  DIRECTION 1 #
+(-)()(U1+U2+U3)(10203;0)(A3010)()
+()()(RHO)(1;0)(A3010)(UOR)+()()(RHO)(2;0)(A3010)(VOR)
+()()(RHO)(3;0)(A3010)(WOR)
+(HBJ)()(HB1,RHO)(112233;0)(A3110)(DETC,UMOR)
+(HBJ)()(UMHB)(112233;0)(A3011)(DETC)
 RHO RHO  3  1 #
()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
+(-)()(RHO)(1;0)(A3010)(UOR)+(-)()(RHO)(1;0)(A31P1)(UOR)
+(-)()(RHO)(2;0)(A3010)(VOR)+(-)()(RHO)(2;0)(A31P1)(VOR)
+(-)()(RHO)(3;0)(A3010)(WOR)+(-)()(RHO)(3;0)(A31P1)(WOR)
 RHO RHO  4  1 #
()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
+(-)()(RHO)(1;0)(A3010)(UOR)+(-)()(RHO)(1;0)(A31P1)(UOR)
+(-)()(RHO)(2;0)(A3010)(VOR)+(-)()(RHO)(2;0)(A31P1)(VOR)
+(-)()(RHO)(3;0)(A3010)(WOR)+(-)()(RHO)(3;0)(A31P1)(WOR)

 ETOT ETOT 1 1 #
()()()(;0)(A200)(DETJ)
 ETOT ETOT 2 1 #
(-)()(U1+U2+U3)(10203;0)(A3010)()
+(-GAM)()(U1+U2+U3)(10203;0)(A3010)()
+(ZPEC,PEI)()()(112233;-1)(A211)()
+(HBJ)()(UMHB)(112233;0)(A3011)(DETC)
 ETOT ETOT 3  1 #
()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
+(GAM)()(U1+U2+U3)(10203;0)(A3010)()+(GAM)()(U1+U2+U3)(10203;0)(A31P1)()
 ETOT ETOT 4  1 #
()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
+(GAM)()(U1+U2+U3)(10203;0)(A3010)()+(GAM)()(U1+U2+U3)(10203;0)(A31P1)()

 M1 M1 1 1 #
()()()(;0)(A200)(DETJ)
 M1 M1 2 1 #
+(-GMH)()()(1;0)(A201)(U1)
+(-)()(M1)(1;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(10203;0)(A3010)()
+(PDUM2,REI)()(OSRH)(112233;-1)(A3011)()
+(HBJ)()(HB1,M1)(112233;0)(A3110)(DETC,UORU)
+(HBJ)()(UMHB)(112233;0)(A3011)(DETC)
 M1 M1 3 1 #
()()(M1)(1;0)(A3010)(OSRH)+()()(M1)(1;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
 M1 M1 4 1 #
(FX2,GMH)()()(1;0)(A201)(U1)
+()()(M1)(1;0)(A3010)(OSRH)+()()(M1)(1;0)(A31P1)(OSRH)
```

```
+()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
 M1  M1  8  1  #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 M2  M2  1   1  #
()()()(;0)(A200)(DETJ)
 M2  M2  2   1  #
+(-GMH)()()(2;0)(A201)(U2)
+(-)()(M2)(2;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(10203;0)(A3010)()
+(PDUM2,REI)()(OSRH)(112233;-1)(A3011)()
+(HBJ)()(HB1,M2)(112233;0)(A3110)(DETC,VORU)
+(HBJ)()(UMHB)(112233;0)(A3011)(DETC)
 M2  M2   3   1  #
()()(M2)(2;0)(A3010)(OSRH)+()()(M2)(2;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
 M2  M2   4   1  #
(FX2,GMH)()()(2;0)(A201)(U2)
+()()(M2)(2;0)(A3010)(OSRH)+()()(M2)(2;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
 M2  M2   8   1  #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 M3  M3  1   1  #
()()()(;0)(A200)(DETJ)
 M3  M3  2   1  #
+(-GMH)()()(3;0)(A201)(U3)
+(-)()(M3)(3;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(10203;0)(A3010)()
+(PDUM2,REI)()(OSRH)(112233;-1)(A3011)()
+(HBJ)()(HB1,M3)(112233;0)(A3110)(DETC,WORU)
+(HBJ)()(UMHB)(112233;0)(A3011)(DETC)
 M3  M3  3   1  #
()()(M3)(3;0)(A3010)(OSRH)+()()(M3)(3;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
 M3  M3  4   1  #
(FX2,GMH)()()(3;0)(A201)(U3)
+()()(M3)(3;0)(A3010)(OSRH)+()()(M3)(3;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(10203;0)(A3010)()+()()(U1+U2+U3)(10203;0)(A31P1)()
 M3  M3  8   1  #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 RHO  M1  2   1  #
(-)()(RHO)(1;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(112233;0)(A3110)(DETC,UORU)
 RHO  M1  3   1  #
()()(RHO)(1;0)(A3010)(OSRH)+()()(RHO)(1;0)(A31P1)(OSRH)
 RHO  M1  4   1  #
()()(RHO)(1;0)(A3010)(OSRH)+()()(RHO)(1;0)(A31P1)(OSRH)

 RHO  M2  2   1  #
(-)()(RHO)(2;0)(A3010)(OSRH)
```

```
+(HBJ)()(HB1,RHO)(112233;0)(A3110)(DETC,VORU)
 RHO M2 3 1 #
()()(RHO)(2;0)(A3010)(OSRH)+()()(RHO)(2;0)(A31P1)(OSRH)
 RHO M2 4 1 #
()()(RHO)(2;0)(A3010)(OSRH)+()()(RHO)(2;0)(A31P1)(OSRH)

 RHO M3 2 1 #
(-)()(RHO)(3;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(112233;0)(A3110)(DETC,WORU)
 RHO M3 3 1 #
()()(RHO)(3;0)(A3010)(OSRH)+()()(RHO)(3;0)(A31P1)(OSRH)
 RHO M3 4 1 #
()()(RHO)(3;0)(A3010)(OSRH)+()()(RHO)(3;0)(A31P1)(OSRH)

 ETOT RHO 2 1 #
()()(EP)(1;0)(A3010)(UOR)+()()(EP)(2;0)(A3010)(VOR)
+()()(EP)(3;0)(A3010)(WOR)
+(HGMMKI)()(U1+U2+U3)(10203;0)(A3010)(UVWS)
+(HBJ)()(HB1,ETOT)(112233;0)(A3110)(DETC,UMOR)
 ETOT RHO 3 1 #
(-)()(EP)(1;0)(A3010)(UOR)+(-)()(EP)(1;0)(A31P1)(UOR)
+(-)()(EP)(2;0)(A3010)(VOR)+(-)()(EP)(2;0)(A31P1)(VOR)
+(-)()(EP)(3;0)(A3010)(WOR)+(-)()(EP)(3;0)(A31P1)(WOR)
+(-,HGMMKI)()(U1+U2+U3)(10203;0)(A3010)(UVWS)
+(-,HGMMKI)()(U1+U2+U3)(10203;0)(A31P1)(UVWS)
 ETOT RHO 4 1 #
(-)()(EP)(1;0)(A3010)(UOR)+(-)()(EP)(1;0)(A31P1)(UOR)
+(-)()(EP)(2;0)(A3010)(VOR)+(-)()(EP)(2;0)(A31P1)(VOR)
+(-)()(EP)(3;0)(A3010)(WOR)+(-)()(EP)(3;0)(A31P1)(WOR)
+(-,HGMMKI)()(U1+U2+U3)(10203;0)(A3010)(UVWS)
+(NF2,-,HGMMKI)()(U1+U2+U3)(10203;0)(A31P1)(UVWS)

 ETOT M1 2 1 #
(-)()(EP)(1;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(112233;0)(A3110)(DETC,UORU)
+(GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U1)
 ETOT M1 3 1 #
()()(EP)(1;0)(A3010)(OSRH)+()()(EP)(1;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U1)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A31P1)(U1)
 ETOT M1 4 1 #
()()(EP)(1;0)(A3010)(OSRH)+()()(EP)(1;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U1)
+(NF2,-GMMKI)()(U1+U2+U3)(10203;0)(A31P1)(U1)

 ETOT M2 2 1 #
(-)()(EP)(2;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(112233;0)(A3110)(DETC,VORU)
+(GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U2)
 ETOT M2 3 1 #
()()(EP)(2;0)(A3010)(OSRH)+()()(EP)(2;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U2)
```

```
+(-GMMKI)()(U1+U2+U3)(10203;0)(A31P1)(U2)
 ETOT M2 4 1 #
 ()()(EP)(2;0)(A3010)(OSRH)+()()(EP)(2;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U2)
+(NF2,-GMMKI)()(U1+U2+U3)(10203;0)(A31P1)(U2)

 ETOT M3 2 1 #
 (-)()(EP)(3;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(112233;0)(A3110)(DETC,WORU)
+(GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U3)
 ETOT M3 3 1 #
 ()()(EP)(3;0)(A3010)(OSRH)+()()(EP)(3;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U3)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A31P1)(U3)
 ETOT M3 4 1 #
 ()()(EP)(3;0)(A3010)(OSRH)+()()(EP)(3;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(10203;0)(A3010)(U3)
+(NF2,-GMMKI)()(U1+U2+U3)(10203;0)(A31P1)(U3)

 M1 RHO 2 1 #
 (GMH)()()(1;0)(A201)(UVWS)
+()()(M1)(1;0)(A3010)(UOR)+()()(M1)(2;0)(A3010)(VOR)
+()()(M1)(3;0)(A3010)(WOR)
+(HBJ)()(HB1,M1)(112233;0)(A3110)(DETC,UMOR)
 M1 RHO 3 1 #
 (-)()(M1)(1;0)(A3010)(UOR)+(-)()(M1)(1;0)(A31P1)(UOR)
+(-)()(M1)(2;0)(A3010)(VOR)+(-)()(M1)(2;0)(A31P1)(VOR)
+(-)()(M1)(3;0)(A3010)(WOR)+(-)()(M1)(3;0)(A31P1)(WOR)
 M1 RHO 4 1 #
 (FX2,-GMH)()()(1;0)(A201)(UVWS)
+(-)()(M1)(1;0)(A3010)(UOR)+(-)()(M1)(1;0)(A31P1)(UOR)
+(-)()(M1)(2;0)(A3010)(VOR)+(-)()(M1)(2;0)(A31P1)(VOR)
+(-)()(M1)(3;0)(A3010)(WOR)+(-)()(M1)(3;0)(A31P1)(WOR)

 M1 ETOT 2 1 #
 (GM1MK)()()(1;0)(A201)()
 M1 ETOT 4 1 #
 (FX2,-GM1MK)()()(1;0)(A201)()

 M1 M2 2 1 #
 (-GM1)()()(1;0)(A201)(U2)
+(-)()(M1)(2;0)(A3010)(OSRH)
+(HBJ)()(HB1,M1)(112233;0)(A3110)(DETC,VORU)
 M1 M2 3 1 #
 ()()(M1)(2;0)(A3010)(OSRH)+()()(M1)(2;0)(A31P1)(OSRH)
 M1 M2 4 1 #
 (FX2,GM1)()()(1;0)(A201)(U2)
+()()(M1)(2;0)(A3010)(OSRH)+()()(M1)(2;0)(A31P1)(OSRH)

 M1 M3 2 1 #
 (-GM1)()()(1;0)(A201)(U3)
+(-)()(M1)(3;0)(A3010)(OSRH)
```

```
+(HBJ)()(HB1,M1)(112233;0)(A3110)(DETC,WORU)
 M1 M3 3 1 #
()()(M1)(3;0)(A3010)(OSRH)+()()(M1)(3;0)(A31P1)(OSRH)
 M1 M3 4 1 #
(FX2,GM1)()()(1;0)(A201)(U3)
+()()(M1)(3;0)(A3010)(OSRH)+()()(M1)(3;0)(A31P1)(OSRH)

 M2 RHO 2 1 #
(GMH)()()(2;0)(A201)(UVWS)
+()()(M2)(1;0)(A3010)(UOR)+()()(M2)(2;0)(A3010)(VOR)
+()()(M2)(3;0)(A3010)(WOR)
+(HBJ)()(HB1,M2)(112233;0)(A3110)(DETC,UMOR)
 M2 RHO 3 1 #
(-)()(M2)(1;0)(A3010)(UOR)+(-)()(M2)(1;0)(A31P1)(UOR)
+(-)()(M2)(2;0)(A3010)(VOR)+(-)()(M2)(2;0)(A31P1)(VOR)
+(-)()(M2)(3;0)(A3010)(WOR)+(-)()(M2)(3;0)(A31P1)(WOR)
 M2 RHO 4 1 #
(FX2,-GMH)()()(2;0)(A201)(UVWS)
+(-)()(M2)(1;0)(A3010)(UOR)+(-)()(M2)(1;0)(A31P1)(UOR)
+(-)()(M2)(2;0)(A3010)(VOR)+(-)()(M2)(2;0)(A31P1)(VOR)
+(-)()(M2)(3;0)(A3010)(WOR)+(-)()(M2)(3;0)(A31P1)(WOR)

 M2 ETOT 2 1 #
(GM1MK)()()(2;0)(A201)()
 M2 ETOT 4 1 #
(FX2,-GM1MK)()()(2;0)(A201)()

 M2 M1  2 1 #
(-GM1)()()(2;0)(A201)(U1)
+(-)()(M2)(1;0)(A3010)(OSRH)
+(HBJ)()(HB1,M2)(112233;0)(A3110)(DETC,UORU)
 M2 M1  3 1 #
()()(M2)(1;0)(A3010)(OSRH)+()()(M2)(1;0)(A31P1)(OSRH)
 M2 M1  4 1 #
(FX2,GM1)()()(2;0)(A201)(U1)
+()()(M2)(1;0)(A3010)(OSRH)+()()(M2)(1;0)(A31P1)(OSRH)

 M2 M3 2 1 #
(-GM1)()()(2;0)(A201)(U3)
+(-)()(M2)(3;0)(A3010)(OSRH)
+(HBJ)()(HB1,M2)(112233;0)(A3110)(DETC,WORU)
 M2 M3 3 1 #
()()(M2)(3;0)(A3010)(OSRH)+()()(M2)(3;0)(A31P1)(OSRH)
 M2 M3 4 1 #
(FX2,GM1)()()(2;0)(A201)(U3)
+()()(M2)(3;0)(A3010)(OSRH)+()()(M2)(3;0)(A31P1)(OSRH)

 M3 RHO 2 1 #
(GMH)()()(3;0)(A201)(UVWS)
+()()(M3)(1;0)(A3010)(UOR)+()()(M3)(2;0)(A3010)(VOR)
+()()(M3)(3;0)(A3010)(WOR)
+(HBJ)()(HB1,M3)(112233;0)(A3110)(DETC,UMOR)
```

```
 M3  RHO  3  1  #
(-)()(M3)(1;0)(A3010)(UOR)+(-)()(M3)(1;0)(A31P1)(UOR)
+(-)()(M3)(2;0)(A3010)(VOR)+(-)()(M3)(2;0)(A31P1)(VOR)
+(-)()(M3)(3;0)(A3010)(WOR)+(-)()(M3)(3;0)(A31P1)(WOR)
 M3  RHO  4  1  #
(FX2,-GMH)()()(3;0)(A201)(UVWS)
+(-)()(M3)(1;0)(A3010)(UOR)+(-)()(M3)(1;0)(A31P1)(UOR)
+(-)()(M3)(2;0)(A3010)(VOR)+(-)()(M3)(2;0)(A31P1)(VOR)
+(-)()(M3)(3;0)(A3010)(WOR)+(-)()(M3)(3;0)(A31P1)(WOR)

 M3  ETOT  2   1  #
(GM1MK)()()(3;0)(A201)()
 M3 ETOT  4    1  #
(FX2,-GM1MK)()()(3;0)(A201)()

 M3  M1  2   1  #
(-GM1)()()(3;0)(A201)(U1)
+(-)()(M3)(1;0)(A3010)(OSRH)
+(HBJ)()(HB1,M3)(112233;0)(A3110)(DETC,UORU)
 M3  M1  3  1  #
()()(M3)(1;0)(A3010)(OSRH)+()()(M3)(1;0)(A31P1)(OSRH)
 M3  M1  4  1  #
(FX2,GM1)()()(3;0)(A201)(U1)
+()()(M3)(1;0)(A3010)(OSRH)+()()(M3)(1;0)(A31P1)(OSRH)

 M3  M2  2  1  #
(-GM1)()()(3;0)(A201)(U2)
+(-)()(M3)(2;0)(A3010)(OSRH)
+(HBJ)()(HB1,M3)(112233;0)(A3110)(DETC,VORU)
 M3  M2  3  1  #
()()(M3)(2;0)(A3010)(OSRH)+()()(M3)(2;0)(A31P1)(OSRH)
 M3  M2  4  1  #
(FX2,GM1)()()(3;0)(A201)(U2)
+()()(M3)(2;0)(A3010)(OSRH)+()()(M3)(2;0)(A31P1)(OSRH)

 #
 #  FACTORED JACOBIAN FOR DIRECTION 2  #
 #

 RHO RHO  1   2  #  VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2
()()()(;0)(A200)(DETJ)
 RHO RHO  2   2  #  VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2
+(-)()(U1+U2+U3)(40506;0)(A3010)()
+()()(RHO)(4;0)(A3010)(UOR)+()()(RHO)(5;0)(A3010)(VOR)
+()()(RHO)(6;0)(A3010)(WOR)
+(HBJ)()(HB1,RHO)(445566;0)(A3110)(DETC,UMOR)
+(HBJ)()(UMHB)(445566;0)(A3011)(DETC)
 RHO RHO  3   2  #  VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2
()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
+(-)()(RHO)(4;0)(A3010)(UOR)+(-)()(RHO)(4;0)(A31P1)(UOR)
+(-)()(RHO)(5;0)(A3010)(VOR)+(-)()(RHO)(5;0)(A31P1)(VOR)
+(-)()(RHO)(6;0)(A3010)(WOR)+(-)()(RHO)(6;0)(A31P1)(WOR)
```

```
 RHO RHO  4  2  #  VARBL, VARDIF, SET, # OF TERMS, DIRECTION 2
 ()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
+(-)()(RHO)(4;0)(A3010)(UOR)+(-)()(RHO)(4;0)(A31P1)(UOR)
+(-)()(RHO)(5;0)(A3010)(VOR)+(-)()(RHO)(5;0)(A31P1)(VOR)
+(-)()(RHO)(6;0)(A3010)(WOR)+(-)()(RHO)(6;0)(A31P1)(WOR)

 ETOT ETOT  1  2  #
 ()()()(;0)(A200)(DETJ)
 ETOT ETOT  2  2  #
 (-)()(U1+U2+U3)(40506;0)(A3010)()
+(-GAM)()(U1+U2+U3)(40506;0)(A3010)()
+(ZPEC,PEI)()()(445566;-1)(A211)()
+(HBJ)()(UMHB)(445566;0)(A3011)(DETC)
 ETOT ETOT  3  2  #
 ()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
+(GAM)()(U1+U2+U3)(40506;0)(A3010)()+(GAM)()(U1+U2+U3)(40506;0)(A31P1)()
 ETOT ETOT  4  2  #
 ()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
+(GAM)()(U1+U2+U3)(40506;0)(A3010)()+(GAM)()(U1+U2+U3)(40506;0)(A31P1)()

 M1  M1  1  2  #
 ()()()(;0)(A200)(DETJ)
 M1  M1  2  2  #
+(-GMH)()()(4;0)(A201)(U1)
+(-)()(M1)(4;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(40506;0)(A3010)()
+(PDUM2,REI)()(OSRH)(445566;-1)(A3011)()
+(HBJ)()(HB1,M1)(445566;0)(A3110)(DETC,UORU)
+(HBJ)()(UMHB)(445566;0)(A3011)(DETC)
 M1  M1  3  2  #
 ()()(M1)(4;0)(A3010)(OSRH)+()()(M1)(4;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
 M1  M1  4  2  #
 (FX2,GMH)()()(4;0)(A201)(U1)
+()()(M1)(4;0)(A3010)(OSRH)+()()(M1)(4;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
 M1  M1  8  2  #
 (PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 M2  M2  1  2  #
 ()()()(;0)(A200)(DETJ)
 M2  M2  2  2  #
+(-GMH)()()(5;0)(A201)(U2)
+(-)()(M2)(5;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(40506;0)(A3010)()
+(PDUM2,REI)()(OSRH)(445566;-1)(A3011)()
+(HBJ)()(HB1,M2)(445566;0)(A3110)(DETC,VORU)
+(HBJ)()(UMHB)(445566;0)(A3011)(DETC)
 M2  M2  3  2  #
 ()()(M2)(5;0)(A3010)(OSRH)+()()(M2)(5;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
 M2  M2  4  2  #
```

```
(FX2,GMH)()()(5;0)(A201)(U2)
+()()(M2)(5;0)(A3010)(OSRH)+()()(M2)(5;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
 M2  M2  8  2  #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 M3  M3  1  2  #
()()()(;0)(A200)(DETJ)
 M3  M3  2  2  #
+(-GMH)()()(6;0)(A201)(U3)
+(-)()(M3)(6;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(40506;0)(A3010)()
+(PDUM2,REI)()(OSRH)(445566;-1)(A3011)()
+(HBJ)()(HB1,M3)(445566;0)(A3110)(DETC,WORU)
+(HBJ)()(UMHB)(445566;0)(A3011)(DETC)
 M3  M3  3  2  #
()()(M3)(6;0)(A3010)(OSRH)+()()(M3)(6;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
 M3  M3  4  2  #
(FX2,GMH)()()(6;0)(A201)(U3)
+()()(M3)(6;0)(A3010)(OSRH)+()()(M3)(6;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(40506;0)(A3010)()+()()(U1+U2+U3)(40506;0)(A31P1)()
 M3  M3  8  2  #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 RHO  M1  2  2  #
(-)()(RHO)(4;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(445566;0)(A3110)(DETC,UORU)
 RHO  M1  3  2  #
()()(RHO)(4;0)(A3010)(OSRH)+()()(RHO)(4;0)(A31P1)(OSRH)
 RHO  M1  4  2  #
()()(RHO)(4;0)(A3010)(OSRH)+()()(RHO)(4;0)(A31P1)(OSRH)

 RHO  M2  2  2  #
(-)()(RHO)(5;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(445566;0)(A3110)(DETC,VORU)
 RHO  M2  3  2  #
()()(RHO)(5;0)(A3010)(OSRH)+()()(RHO)(5;0)(A31P1)(OSRH)
 RHO  M2  4  2  #
()()(RHO)(5;0)(A3010)(OSRH)+()()(RHO)(5;0)(A31P1)(OSRH)

 RHO  M3  2  2  #
(-)()(RHO)(6;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(445566;0)(A3110)(DETC,WORU)
 RHO  M3  3  2  #
()()(RHO)(6;0)(A3010)(OSRH)+()()(RHO)(6;0)(A31P1)(OSRH)
 RHO  M3  4  2  #
()()(RHO)(6;0)(A3010)(OSRH)+()()(RHO)(6;0)(A31P1)(OSRH)

 ETOT RHO  2  2  #
()()(EP)(4;0)(A3010)(UOR)+()()(EP)(5;0)(A3010)(VOR)
+()()(EP)(6;0)(A3010)(WOR)
```

```
+(HGMMKI)()(U1+U2+U3)(40506;0)(A3010)(UVWS)
+(HBJ)()(HB1,ETOT)(445566;0)(A3110)(DETC,UMOR)
 ETOT RHO 3 2 #
(-)()(EP)(4;0)(A3010)(UOR)+(-)()(EP)(4;0)(A31P1)(UOR)
+(-)()(EP)(5;0)(A3010)(VOR)+(-)()(EP)(5;0)(A31P1)(VOR)
+(-)()(EP)(6;0)(A3010)(WOR)+(-)()(EP)(6;0)(A31P1)(WOR)
+(-,HGMMKI)()(U1+U2+U3)(40506;0)(A3010)(UVWS)
+(-,HGMMKI)()(U1+U2+U3)(40506;0)(A31P1)(UVWS)
 ETOT RHO 4 2 #
(-)()(EP)(4;0)(A3010)(UOR)+(-)()(EP)(4;0)(A31P1)(UOR)
+(-)()(EP)(5;0)(A3010)(VOR)+(-)()(EP)(5;0)(A31P1)(VOR)
+(-)()(EP)(6;0)(A3010)(WOR)+(-)()(EP)(6;0)(A31P1)(WOR)
+(-,HGMMKI)()(U1+U2+U3)(40506;0)(A3010)(UVWS)
+(NF2,-,HGMMKI)()(U1+U2+U3)(40506;0)(A31P1)(UVWS)

 ETOT M1 2 2 #
(-)()(EP)(4;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(445566;0)(A3110)(DETC,UORU)
+(GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U1)
 ETOT M1 3 2 #
()()(EP)(4;0)(A3010)(OSRH)+()()(EP)(4;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U1)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A31P1)(U1)
 ETOT M1 4 2 #
()()(EP)(4;0)(A3010)(OSRH)+()()(EP)(4;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U1)
+(NF2,-GMMKI)()(U1+U2+U3)(40506;0)(A31P1)(U1)

 ETOT M2 2 2 #
(-)()(EP)(5;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(445566;0)(A3110)(DETC,VORU)
+(GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U2)
 ETOT M2 3 2 #
()()(EP)(5;0)(A3010)(OSRH)+()()(EP)(5;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U2)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A31P1)(U2)
 ETOT M2 4 2 #
()()(EP)(5;0)(A3010)(OSRH)+()()(EP)(5;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U2)
+(NF2,-GMMKI)()(U1+U2+U3)(40506;0)(A31P1)(U2)

 ETOT M3 2 2 #
(-)()(EP)(6;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(445566;0)(A3110)(DETC,WORU)
+(GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U3)
 ETOT M3 3 2 #
()()(EP)(6;0)(A3010)(OSRH)+()()(EP)(6;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U3)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A31P1)(U3)
 ETOT M3 4 2 #
()()(EP)(6;0)(A3010)(OSRH)+()()(EP)(6;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(40506;0)(A3010)(U3)
```

```
+(NF2,-GMMKI)()(U1+U2+U3)(40506;0)(A31P1)(U3)

 M1 RHO 2 2 #
(GMH)()()(4;0)(A201)(UVWS)
+()()(M1)(4;0)(A3010)(UOR)+()()(M1)(5;0)(A3010)(VOR)
+()()(M1)(6;0)(A3010)(WOR)
+(HBJ)()(HB1,M1)(445566;0)(A3110)(DETC,UMOR)
 M1 RHO 3 2 #
(-)()(M1)(4;0)(A3010)(UOR)+(-)()(M1)(4;0)(A31P1)(UOR)
+(-)()(M1)(5;0)(A3010)(VOR)+(-)()(M1)(5;0)(A31P1)(VOR)
+(-)()(M1)(6;0)(A3010)(WOR)+(-)()(M1)(6;0)(A31P1)(WOR)
 M1 RHO 4 2 #
(FX2,-GMH)()()(4;0)(A201)(UVWS)
+(-)()(M1)(4;0)(A3010)(UOR)+(-)()(M1)(4;0)(A31P1)(UOR)
+(-)()(M1)(5;0)(A3010)(VOR)+(-)()(M1)(5;0)(A31P1)(VOR)
+(-)()(M1)(6;0)(A3010)(WOR)+(-)()(M1)(6;0)(A31P1)(WOR)

 M1 ETOT 2 2 #
(GM1MK)()()(4;0)(A201)()
 M1 ETOT 4 2 #
(FX2,-GM1MK)()()(4;0)(A201)()

 M1 M2 2 2 #
(-GM1)()()(4;0)(A201)(U2)
+(-)()(M1)(5;0)(A3010)(OSRH)
+(HBJ)()(HB1,M1)(445566;0)(A3110)(DETC,VORU)
 M1 M2 3 2 #
()()(M1)(5;0)(A3010)(OSRH)+()()(M1)(5;0)(A31P1)(OSRH)
 M1 M2 4 2 #
(FX2,GM1)()()(4;0)(A201)(U2)
+()()(M1)(5;0)(A3010)(OSRH)+()()(M1)(5;0)(A31P1)(OSRH)

 M1 M3 2 2 #
(-GM1)()()(4;0)(A201)(U3)
+(-)()(M1)(6;0)(A3010)(OSRH)
+(HBJ)()(HB1,M1)(445566;0)(A3110)(DETC,WORU)
 M1 M3 3 2 #
()()(M1)(6;0)(A3010)(OSRH)+()()(M1)(6;0)(A31P1)(OSRH)
 M1 M3 4 2 #
(FX2,GM1)()()(4;0)(A201)(U3)
+()()(M1)(6;0)(A3010)(OSRH)+()()(M1)(6;0)(A31P1)(OSRH)

 M2 RHO 2 2 #
(GMH)()()(5;0)(A201)(UVWS)
+()()(M2)(4;0)(A3010)(UOR)+()()(M2)(5;0)(A3010)(VOR)
+()()(M2)(6;0)(A3010)(WOR)
+(HBJ)()(HB1,M2)(445566;0)(A3110)(DETC,UMOR)
 M2 RHO 3 2 #
(-)()(M2)(4;0)(A3010)(UOR)+(-)()(M2)(4;0)(A31P1)(UOR)
+(-)()(M2)(5;0)(A3010)(VOR)+(-)()(M2)(5;0)(A31P1)(VOR)
+(-)()(M2)(6;0)(A3010)(WOR)+(-)()(M2)(6;0)(A31P1)(WOR)
 M2 RHO 4 2 #
```

```
(FX2,-GMH)()()(5;0)(A201)(UVWS)
+(-)()(M2)(4;0)(A3010)(UOR)+(-)()(M2)(4;0)(A31P1)(UOR)
+(-)()(M2)(5;0)(A3010)(VOR)+(-)()(M2)(5;0)(A31P1)(VOR)
+(-)()(M2)(6;0)(A3010)(WOR)+(-)()(M2)(6;0)(A31P1)(WOR)

 M2 ETOT  2  2  #
(GM1MK)()()(5;0)(A201)()
 M2 ETOT  4  2  #
(FX2,-GM1MK)()()(5;0)(A201)()

 M2 M1   2  2  #
(-GM1)()()(5;0)(A201)(U1)
+(-)()(M2)(4;0)(A3010)(OSRH)
+(HBJ)()(HB1,M2)(445566;0)(A3110)(DETC,UORU)
 M2  M1  3  2  #
()()(M2)(4;0)(A3010)(OSRH)+()()(M2)(4;0)(A31P1)(OSRH)
 M2  M1  4  2  #
(FX2,GM1)()()(5;0)(A201)(U1)
+()()(M2)(4;0)(A3010)(OSRH)+()()(M2)(4;0)(A31P1)(OSRH)

 M2  M3  2  2  #
(-GM1)()()(5;0)(A201)(U3)
+(-)()(M2)(6;0)(A3010)(OSRH)
+(HBJ)()(HB1,M2)(445566;0)(A3110)(DETC,WORU)
 M2  M3  3  2  #
()()(M2)(6;0)(A3010)(OSRH)+()()(M2)(6;0)(A31P1)(OSRH)
 M2  M3  4  2  #
(FX2,GM1)()()(5;0)(A201)(U3)
+()()(M2)(6;0)(A3010)(OSRH)+()()(M2)(6;0)(A31P1)(OSRH)

 M3  RHO  2  2  #
(GMH)()()(6;0)(A201)(UVWS)
+()()(M3)(4;0)(A3010)(UOR)+()()(M3)(5;0)(A3010)(VOR)
+()()(M3)(6;0)(A3010)(WOR)
+(HBJ)()(HB1,M3)(445566;0)(A3110)(DETC,UMOR)
 M3  RHO  3  2  #
(-)()(M3)(4;0)(A3010)(UOR)+(-)()(M3)(4;0)(A31P1)(UOR)
+(-)()(M3)(5;0)(A3010)(VOR)+(-)()(M3)(5;0)(A31P1)(VOR)
+(-)()(M3)(6;0)(A3010)(WOR)+(-)()(M3)(6;0)(A31P1)(WOR)
 M3  RHO  4  2  #
(FX2,-GMH)()()(6;0)(A201)(UVWS)
+(-)()(M3)(4;0)(A3010)(UOR)+(-)()(M3)(4;0)(A31P1)(UOR)
+(-)()(M3)(5;0)(A3010)(VOR)+(-)()(M3)(5;0)(A31P1)(VOR)
+(-)()(M3)(6;0)(A3010)(WOR)+(-)()(M3)(6;0)(A31P1)(WOR)

 M3  ETOT  2  2  #
(GM1MK)()()(6;0)(A201)()
 M3  ETOT  4  2  #
(FX2,-GM1MK)()()(6;0)(A201)()

 M3  M1  2  2  #
(-GM1)()()(6;0)(A201)(U1)
```

```
+(-)()(M3)(4;0)(A3010)(OSRH)
+(HBJ)()(HB1,M3)(445566;0)(A3110)(DETC,UORU)
 M3  M1  3  2  #
()()(M3)(4;0)(A3010)(OSRH)+()()(M3)(4;0)(A31P1)(OSRH)
 M3  M1  4  2  #
(FX2,GM1)()()(6;0)(A201)(U1)
+()()(M3)(4;0)(A3010)(OSRH)+()()(M3)(4;0)(A31P1)(OSRH)

 M3  M2  2  2  #
(-GM1)()()(6;0)(A201)(U2)
+(-)()(M3)(5;0)(A3010)(OSRH)
+(HBJ)()(HB1,M3)(445566;0)(A3110)(DETC,VORU)
 M3  M2  3  2  #
()()(M3)(5;0)(A3010)(OSRH)+()()(M3)(5;0)(A31P1)(OSRH)
 M3  M2  4  2  #
(FX2,GM1)()()(6;0)(A201)(U2)
+()()(M3)(5;0)(A3010)(OSRH)+()()(M3)(5;0)(A31P1)(OSRH)

 #
 #  FACTORED JACOBIAN FOR DIRECTION 3  #
 #

RHO RHO 1  3  #  VARBL, VARDIF, SET,  DIRECTION 3  #
()()()(;0)(A200)(DETJ)
RHO RHO 2  3  #  VARBL, VARDIF, SET,  DIRECTION 3  #
+(-)()(U1+U2+U3)(70809;0)(A3010)()
+()()(RHO)(7;0)(A3010)(UOR)+()()(RHO)(8;0)(A3010)(VOR)
+()()(RHO)(9;0)(A3010)(WOR)
+(HBJ)()(HB1,RHO)(778899;0)(A3110)(DETC,UMOR)
+(HBJ)()(UMHB)(778899;0)(A3011)(DETC)
RHO RHO 3  3  #  VARBL, VARDIF, SET,  DIRECTION 3  #
()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
+(-)()(RHO)(7;0)(A3010)(UOR)+(-)()(RHO)(7;0)(A31P1)(UOR)
+(-)()(RHO)(8;0)(A3010)(VOR)+(-)()(RHO)(8;0)(A31P1)(VOR)
+(-)()(RHO)(9;0)(A3010)(WOR)+(-)()(RHO)(9;0)(A31P1)(WOR)
RHO RHO 4  3  #  VARBL, VARDIF, SET,  DIRECTION 3  #
()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
+(-)()(RHO)(7;0)(A3010)(UOR)+(-)()(RHO)(7;0)(A31P1)(UOR)
+(-)()(RHO)(8;0)(A3010)(VOR)+(-)()(RHO)(8;0)(A31P1)(VOR)
+(-)()(RHO)(9;0)(A3010)(WOR)+(-)()(RHO)(9;0)(A31P1)(WOR)

 ETOT  ETOT  1  3  #
()()()(;0)(A200)(DETJ)
 ETOT  ETOT  2  3  #
+(-)()(U1+U2+U3)(70809;0)(A3010)()
+(-GAM)()(U1+U2+U3)(70809;0)(A3010)()
+(ZPEC,PEI)()()(778899;-1)(A211)()
+(HBJ)()(UMHB)(778899;0)(A3011)(DETC)
 ETOT  ETOT  3  3  #
()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
+(GAM)()(U1+U2+U3)(70809;0)(A3010)()+(GAM)()(U1+U2+U3)(70809;0)(A31P1)()
 ETOT  ETOT  4  3  #
```

```
()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
+(GAM)()(U1+U2+U3)(70809;0)(A3010)()+(GAM)()(U1+U2+U3)(70809;0)(A31P1)()

 M1  M1  1   3  #
()()()(;0)(A200)(DETJ)
 M1  M1  2   3  #
+(-GMH)()()(7;0)(A201)(U1)
+(-)()(M1)(7;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(70809;0)(A3010)()
+(PDUM2,REI)()(OSRH)(778899;-1)(A3011)()
+(HBJ)()(HB1,M1)(778899;0)(A3110)(DETC,UORU)
+(HBJ)()(UMHB)(778899;0)(A3011)(DETC)
 M1  M1  3   3  #
()()(M1)(7;0)(A3010)(OSRH)+()()(M1)(7;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
 M1  M1  4   3  #
(FX2,GMH)()()(7;0)(A201)(U1)
+()()(M1)(7;0)(A3010)(OSRH)+()()(M1)(7;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
 M1  M1  8   3  #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 M2  M2  1   3  #
()()()(;0)(A200)(DETJ)
 M2  M2  2   3  #
+(-GMH)()()(8;0)(A201)(U2)
+(-)()(M2)(8;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(70809;0)(A3010)()
+(PDUM2,REI)()(OSRH)(778899;-1)(A3011)()
+(HBJ)()(HB1,M2)(778899;0)(A3110)(DETC,VORU)
+(HBJ)()(UMHB)(778899;0)(A3011)(DETC)
 M2  M2  3   3  #
()()(M2)(8;0)(A3010)(OSRH)+()()(M2)(8;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
 M2  M2  4   3  #
(FX2,GMH)()()(8;0)(A201)(U2)
+()()(M2)(8;0)(A3010)(OSRH)+()()(M2)(8;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
 M2  M2  8   3  #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 M3  M3  1   3  #
()()()(;0)(A200)(DETJ)
 M3  M3  2   3  #
+(-GMH)()()(9;0)(A201)(U3)
+(-)()(M3)(9;0)(A3010)(OSRH)
+(-)()(U1+U2+U3)(70809;0)(A3010)()
+(PDUM2,REI)()(OSRH)(778899;-1)(A3011)()
+(HBJ)()(HB1,M3)(778899;0)(A3110)(DETC,WORU)
+(HBJ)()(UMHB)(778899;0)(A3011)(DETC)
 M3  M3  3   3  #
()()(M3)(9;0)(A3010)(OSRH)+()()(M3)(9;0)(A31P1)(OSRH)
```

```
+()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
 M3 M3 4 3 #
(FX2,GMH)()()(9;0)(A201)(U3)
+()()(M3)(9;0)(A3010)(OSRH)+()()(M3)(9;0)(A31P1)(OSRH)
+()()(U1+U2+U3)(70809;0)(A3010)()+()()(U1+U2+U3)(70809;0)(A31P1)()
 M3 M3 8 3 #
(PDUM2,REI)()(YPLS)(;0)(A200)(DETJ)

 RHO M1 2 3 #
(-)()(RHO)(7;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(778899;0)(A3110)(DETC,UORU)
 RHO M1 3 3 #
()()(RHO)(7;0)(A3010)(OSRH)+()()(RHO)(7;0)(A31P1)(OSRH)
 RHO M1 4 3 #
()()(RHO)(7;0)(A3010)(OSRH)+()()(RHO)(7;0)(A31P1)(OSRH)

 RHO M2 2 3 #
(-)()(RHO)(8;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(778899;0)(A3110)(DETC,VORU)
 RHO M2 3 3 #
()()(RHO)(8;0)(A3010)(OSRH)+()()(RHO)(8;0)(A31P1)(OSRH)
 RHO M2 4 3 #
()()(RHO)(8;0)(A3010)(OSRH)+()()(RHO)(8;0)(A31P1)(OSRH)

 RHO M3 2 3 #
(-)()(RHO)(9;0)(A3010)(OSRH)
+(HBJ)()(HB1,RHO)(778899;0)(A3110)(DETC,WORU)
 RHO M3 3 3 #
()()(RHO)(9;0)(A3010)(OSRH)+()()(RHO)(9;0)(A31P1)(OSRH)
 RHO M3 4 3 #
()()(RHO)(9;0)(A3010)(OSRH)+()()(RHO)(9;0)(A31P1)(OSRH)

 ETOT RHO 2 3 #
()()(EP)(7;0)(A3010)(UOR)+()()(EP)(8;0)(A3010)(VOR)
+()()(EP)(9;0)(A3010)(WOR)
+(HGMMKI)()(U1+U2+U3)(70809;0)(A3010)(UVWS)
+(HBJ)()(HB1,ETOT)(778899;0)(A3110)(DETC,UMOR)
 ETOT RHO 3 3 #
(-)()(EP)(7;0)(A3010)(UOR)+(-)()(EP)(7;0)(A31P1)(UOR)
+(-)()(EP)(8;0)(A3010)(VOR)+(-)()(EP)(8;0)(A31P1)(VOR)
+(-)()(EP)(9;0)(A3010)(WOR)+(-)()(EP)(9;0)(A31P1)(WOR)
+(-,HGMMKI)()(U1+U2+U3)(70809;0)(A3010)(UVWS)
+(-,HGMMKI)()(U1+U2+U3)(70809;0)(A31P1)(UVWS)
 ETOT RHO 4 3 #
(-)()(EP)(7;0)(A3010)(UOR)+(-)()(EP)(7;0)(A31P1)(UOR)
+(-)()(EP)(8;0)(A3010)(VOR)+(-)()(EP)(8;0)(A31P1)(VOR)
+(-)()(EP)(9;0)(A3010)(WOR)+(-)()(EP)(9;0)(A31P1)(WOR)
+(-,HGMMKI)()(U1+U2+U3)(70809;0)(A3010)(UVWS)
+(NF2,-,HGMMKI)()(U1+U2+U3)(70809;0)(A31P1)(UVWS)

 ETOT M1 2 3 #
(-)()(EP)(7;0)(A3010)(OSRH)
```

```
+(HBJ)()(HB1,ETOT)(778899;0)(A3110)(DETC,UORU)
+(GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U1)
 ETOT M1 3 3 #
 ()()(EP)(7;0)(A3010)(OSRH)+()()(EP)(7;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U1)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A31P1)(U1)
 ETOT M1 4 3 #
 ()()(EP)(7;0)(A3010)(OSRH)+()()(EP)(7;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U1)
+(NF2,-GMMKI)()(U1+U2+U3)(70809;0)(A31P1)(U1)

 ETOT M2 2 3 #
 (-)()(EP)(8;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(778899;0)(A3110)(DETC,VORU)
+(GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U2)
 ETOT M2 3 3 #
 ()()(EP)(8;0)(A3010)(OSRH)+()()(EP)(8;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U2)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A31P1)(U2)
 ETOT M2 4 3 #
 ()()(EP)(8;0)(A3010)(OSRH)+()()(EP)(8;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U2)
+(NF2,-GMMKI)()(U1+U2+U3)(70809;0)(A31P1)(U2)

 ETOT M3 2 3 #
 (-)()(EP)(9;0)(A3010)(OSRH)
+(HBJ)()(HB1,ETOT)(778899;0)(A3110)(DETC,WORU)
+(GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U3)
 ETOT M3 3 3 #
 ()()(EP)(9;0)(A3010)(OSRH)+()()(EP)(9;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U3)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A31P1)(U3)
 ETOT M3 4 3 #
 ()()(EP)(9;0)(A3010)(OSRH)+()()(EP)(9;0)(A31P1)(OSRH)
+(-GMMKI)()(U1+U2+U3)(70809;0)(A3010)(U3)
+(NF2,-GMMKI)()(U1+U2+U3)(70809;0)(A31P1)(U3)

 M1 RHO 2 3 #
 (GMH)()()(7;0)(A201)(UVWS)
+()()(M1)(7;0)(A3010)(UOR)+()()(M1)(8;0)(A3010)(VOR)
+()()(M1)(9;0)(A3010)(WOR)
+(HBJ)()(HB1,M1)(778899;0)(A3110)(DETC,UMOR)
 M1 RHO 3 3 #
 (-)()(M1)(7;0)(A3010)(UOR)+(-)()(M1)(7;0)(A31P1)(UOR)
+(-)()(M1)(8;0)(A3010)(VOR)+(-)()(M1)(8;0)(A31P1)(VOR)
+(-)()(M1)(9;0)(A3010)(WOR)+(-)()(M1)(9;0)(A31P1)(WOR)
 M1 RHO 4 3 #
 (FX2,-GMH)()()(7;0)(A201)(UVWS)
+(-)()(M1)(7;0)(A3010)(UOR)+(-)()(M1)(7;0)(A31P1)(UOR)
+(-)()(M1)(8;0)(A3010)(VOR)+(-)()(M1)(8;0)(A31P1)(VOR)
+(-)()(M1)(9;0)(A3010)(WOR)+(-)()(M1)(9;0)(A31P1)(WOR)
```

```
 M1 ETOT 2  3  #
(GM1MK)()()(7;0)(A201)()
 M1 ETOT 4  3  #
(FX2,-GM1MK)()()(7;0)(A201)()

 M1  M2  2  3  #
(-GM1)()()(7;0)(A201)(U2)
+(-)()(M1)(8;0)(A3010)(OSRH)
+(HBJ)()(HB1,M1)(778899;0)(A3110)(DETC,VORU)
 M1  M2  3  3  #
()()(M1)(8;0)(A3010)(OSRH)+()()(M1)(8;0)(A31P1)(OSRH)
 M1  M2  4  3  #
(FX2,GM1)()()(7;0)(A201)(U2)
+()()(M1)(8;0)(A3010)(OSRH)+()()(M1)(8;0)(A31P1)(OSRH)

 M1  M3  2  3  #
(-GM1)()()(7;0)(A201)(U3)
+(-)()(M1)(9;0)(A3010)(OSRH)
+(HBJ)()(HB1,M1)(778899;0)(A3110)(DETC,WORU)
 M1  M3  3  3  #
()()(M1)(9;0)(A3010)(OSRH)+()()(M1)(9;0)(A31P1)(OSRH)
 M1  M3  4  3  #
(FX2,GM1)()()(7;0)(A201)(U3)
+()()(M1)(9;0)(A3010)(OSRH)+()()(M1)(9;0)(A31P1)(OSRH)

 M2  RHO  2  3  #
(GMH)()()(8;0)(A201)(UVWS)
+()()(M2)(7;0)(A3010)(UOR)+()()(M2)(8;0)(A3010)(VOR)
+()()(M2)(9;0)(A3010)(WOR)
+(HBJ)()(HB1,M2)(778899;0)(A3110)(DETC,UMOR)
 M2  RHO  3  3  #
(-)()(M2)(7;0)(A3010)(UOR)+(-)()(M2)(7;0)(A31P1)(UOR)
+(-)()(M2)(8;0)(A3010)(VOR)+(-)()(M2)(8;0)(A31P1)(VOR)
+(-)()(M2)(9;0)(A3010)(WOR)+(-)()(M2)(9;0)(A31P1)(WOR)
 M2  RHO  4  3  #
(FX2,-GMH)()()(8;0)(A201)(UVWS)
+(-)()(M2)(7;0)(A3010)(UOR)+(-)()(M2)(7;0)(A31P1)(UOR)
+(-)()(M2)(8;0)(A3010)(VOR)+(-)()(M2)(8;0)(A31P1)(VOR)
+(-)()(M2)(9;0)(A3010)(WOR)+(-)()(M2)(9;0)(A31P1)(WOR)

 M2 ETOT  2  3  #
(GM1MK)()()(8;0)(A201)()
 M2 ETOT  4  3  #
(FX2,-GM1MK)()()(8;0)(A201)()

 M2  M1  2  3  #
(-GM1)()()(8;0)(A201)(U1)
+(-)()(M2)(7;0)(A3010)(OSRH)
+(HBJ)()(HB1,M2)(778899;0)(A3110)(DETC,UORU)
 M2  M1  3  3  #
()()(M2)(7;0)(A3010)(OSRH)+()()(M2)(7;0)(A31P1)(OSRH)
 M2  M1  4  3  #
```

```
(FX2,GM1)()()(8;0)(A201)(U1)
+()()(M2)(7;0)(A3010)(OSRH)+()()(M2)(7;0)(A31P1)(OSRH)

 M2 M3 2 3 #
(-GM1)()()(8;0)(A201)(U3)
+(-)()(M2)(9;0)(A3010)(OSRH)
+(HBJ)()(HB1,M2)(778899;0)(A3110)(DETC,WORU)
 M2 M3 3 3 #
()()(M2)(9;0)(A3010)(OSRH)+()()(M2)(9;0)(A31P1)(OSRH)
 M2 M3 4 3 #
(FX2,GM1)()()(8;0)(A201)(U3)
+()()(M2)(9;0)(A3010)(OSRH)+()()(M2)(9;0)(A31P1)(OSRH)

 M3 RHO 2 3 #
(GMH)()()(9;0)(A201)(UVWS)
+()()(M3)(7;0)(A3010)(UOR)+()()(M3)(8;0)(A3010)(VOR)
+()()(M3)(9;0)(A3010)(WOR)
+(HBJ)()(HB1,M3)(778899;0)(A3110)(DETC,UMOR)
 M3 RHO 3 3 #
(-)()(M3)(7;0)(A3010)(UOR)+(-)()(M3)(7;0)(A31P1)(UOR)
+(-)()(M3)(8;0)(A3010)(VOR)+(-)()(M3)(8;0)(A31P1)(VOR)
+(-)()(M3)(9;0)(A3010)(WOR)+(-)()(M3)(9;0)(A31P1)(WOR)
 M3 RHO 4 3 #
(FX2,-GMH)()()(9;0)(A201)(UVWS)
+(-)()(M3)(7;0)(A3010)(UOR)+(-)()(M3)(7;0)(A31P1)(UOR)
+(-)()(M3)(8;0)(A3010)(VOR)+(-)()(M3)(8;0)(A31P1)(VOR)
+(-)()(M3)(9;0)(A3010)(WOR)+(-)()(M3)(9;0)(A31P1)(WOR)

 M3 ETOT 2 3 #
(GM1MK)()()(9;0)(A201)()
 M3 ETOT 4 3 #
(FX2,-GM1MK)()()(9;0)(A201)()

 M3 M1 2 3 #
(-GM1)()()(9;0)(A201)(U1)
+(-)()(M3)(7;0)(A3010)(OSRH)
+(HBJ)()(HB1,M3)(778899;0)(A3110)(DETC,UORU)
 M3 M1 3 3 #
()()(M3)(7;0)(A3010)(OSRH)+()()(M3)(7;0)(A31P1)(OSRH)
 M3 M1 4 3 #
(FX2,GM1)()()(9;0)(A201)(U1)
+()()(M3)(7;0)(A3010)(OSRH)+()()(M3)(7;0)(A31P1)(OSRH)

 M3 M2 2 3 #
(-GM1)()()(9;0)(A201)(U2)
+(-)()(M3)(8;0)(A3010)(OSRH)
+(HBJ)()(HB1,M3)(778899;0)(A3110)(DETC,VORU)
 M3 M2 3 3 #
()()(M3)(8;0)(A3010)(OSRH)+()()(M3)(8;0)(A31P1)(OSRH)
 M3 M2 4 3 #
(FX2,GM1)()()(9;0)(A201)(U2)
+()()(M3)(8;0)(A3010)(OSRH)+()()(M3)(8;0)(A31P1)(OSRH)
```

```
GROUP FREQUENCY
  1
SOLUTION TYPE
 DELTA_Q
 FACTORED_GAUSS_ELIMINATION
 IMPLICIT_EULER

END
```